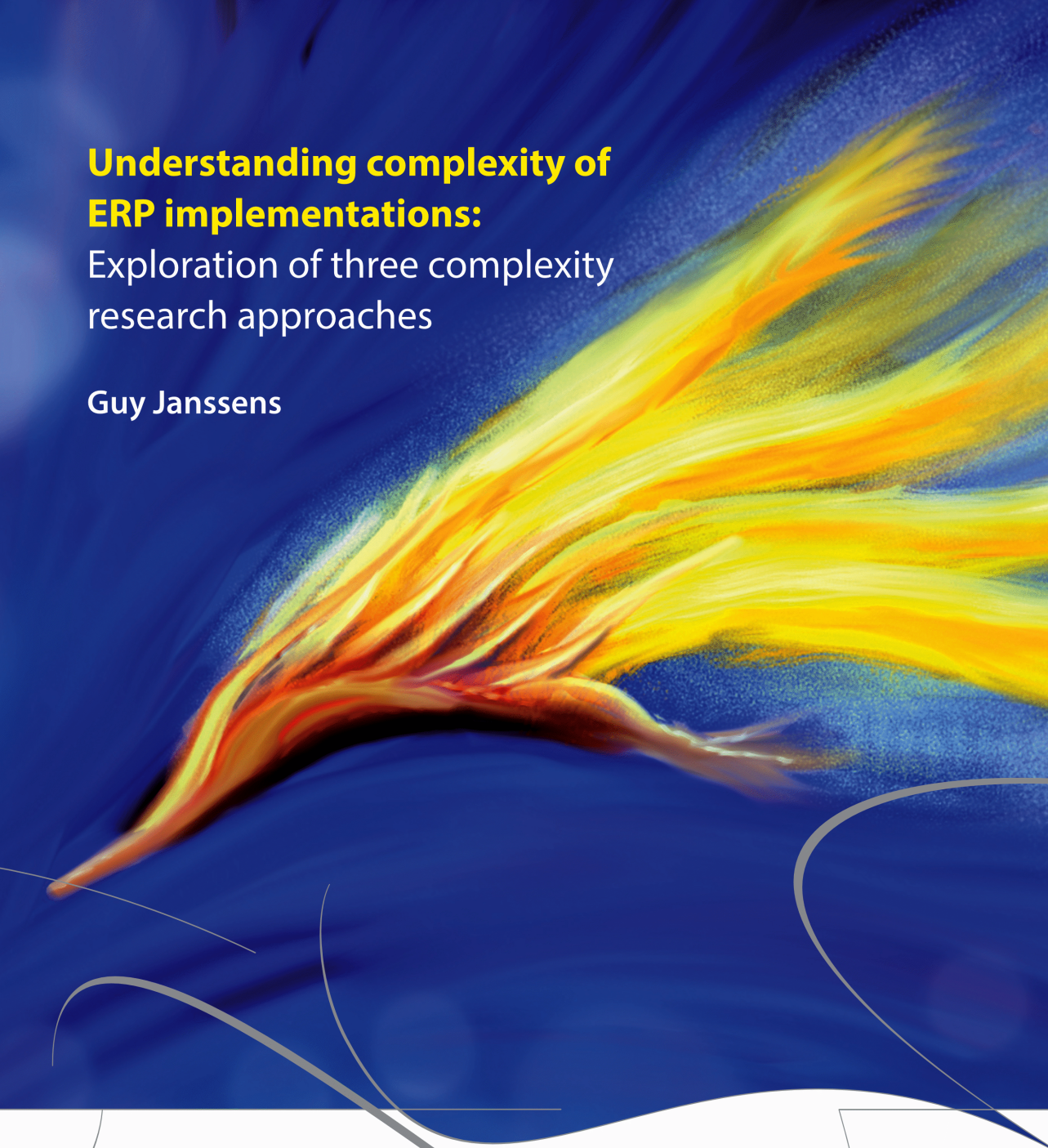


Understanding complexity of ERP implementations:

Exploration of three complexity research approaches

Guy Janssens

Open Universiteit
www.ou.nl

Understanding complexity of ERP implementations: Exploration of three complexity research approaches

Understanding complexity of ERP implementations: Exploration of three complexity research approaches

Proefschrift

ter verkrijging van de graad van doctor
aan de Open Universiteit
op gezag van de rector magnificus
prof. mr. A. Oskamp
ten overstaan van een door het
College voor promoties ingestelde commissie
in het openbaar te verdedigen

op vrijdag 13 oktober 2017 te Heerlen
om 13:30 uur precies

door

Guy Janssens

geboren op 5 augustus 1961 te Heerlen

Promotor
Prof. dr. R.J. Kusters, Open Universiteit

Co-promotor
Dr. ir. H.H. Martin, Open Universiteit

Overige leden beoordelingscommissie
Prof. dr. ir. L. Pintelon, KU Leuven
Prof. dr. P.M.A. Ribbers, Tilburg University
Prof. dr. H.M.J. van den Bosch, Open Universiteit
Prof. dr. ir. J.J.M. Trienekens, Open Universiteit

Cover design: Guy Janssens
Cover Photo: “Chimera”/“Hersenspinsel” (painting by Guy Janssens)
Printed by Datawyse
ISBN: 978 94 92739 01 8

Chimera A thing that is hoped or wished for, but in fact is illusory of impossible to achieve.

CONTENTS

SUMMARY

This summary consists of two parts:

1. The first part encompasses a summary discussing methodology, results and conclusions for the main purpose of this research.
2. The second part encompasses the individual summaries of every performed study.

Summary for the main purpose of this research

Implementing enterprise resource planning (ERP) systems (in this thesis referred to as ERP implementation) is considered to be a complex matter. ERP implementation is considered complex because, in most cases, it affects many parts of an organisation. Also, the costs of software, hardware, maintenance and particularly the implementation process itself are high and risky for an organisation. Therefore, research in ERP implementation is focused on providing practice with useful insights and tools for better cost and risk handling. ERP implementations are executed for over 20 years. Unfortunately, neither science nor practice has produced sufficient tools, guidelines and methods for proper management and control of ERP implementations. After 20 years, research and practice still consider ERP implementation a complex matter. Through our research, we intend to approach this complexity of ERP implementation from a new perspective. A perspective which takes the mere construct 'complexity' as a starting point.
We studied ERP implementation complexity as a construct in the context of complexity research and asked ourselves how the complexity of ERP implementations can best be explored.

Our main research question is:

What is the added value of explicit application of different complexity research approaches into ERP implementation?

We conducted a structured literature study into the use of the construct of complexity in existing ERP implementation research. Although the terms 'complex' and 'complexity' are quite often used in ERP research results, a definition and discussion of the construct of complexity in the context of ERP implementation, is insufficiently addressed. Therefore, we provided a definition of ERP implementation complexity based on complexity literature. In addition, during our literature study, we noticed that we could not detect explicitly or implicitly used research complexity approaches. This lack compelled us to conclude that, to be able to address our research question, we needed a practical differentiation for detecting and discriminating complexity research approaches. Also, we needed this differentiation to be able to determine the value of each approach. Therefore, we chose Manson's three complexity research approaches as the foundation of our research:

- algorithmic complexity research approach
- deterministic complexity research approach
- aggregate complexity research approach

Based on Manson's differentiation and our literature study results, we concluded that a deterministic complexity research approach prevails within mainstream ERP implementation research. We did not encounter an aggregate or algorithmic complexity research approach in the results of our literature study. We did not explore the algorithmic complexity research approach in our empirical research any further because we reasoned that this approach would be too limited when used as a principle for understanding the actual complexity of ERP implementation. However, for the deterministic and aggregate complexity approaches, we conducted empirical research to determine the value of both approaches to research and practice. We conducted three ERP implementation studies by a deterministic complexity research approach and one study by an aggregate complexity research approach. The results of these four studies provided sufficient insights into the requested values of the deterministic and aggregate complexity research approaches. In addition to these insights, these four studies also provided useful results as separate studies.
Based on the results of these four studies we conclude that both deterministic and aggregate complexity research approaches appear valuable, can complement each

other and thus are relevant for research and practice. We conclude that it is valuable to perform more research into ERP implementations explicitly by an aggregate complexity research approach. Conducting more research by an aggregate complexity research approach may well complement the results from research by deterministic complexity approaches, lead to different points of view on ERP implementations and offer new insights for both research and practice.

We consider it important to remain aware of the paradigm on which a complexity research approach is based. Managing an ERP implementation project with exclusively a deterministic complexity paradigm in mind bears a risk of a false sense of security when relying solely on factors and directives derived from deterministic ERP implementation complexity research. Also, top management, members of the ERP implementation project and other stakeholders should be aware of the aggregate paradigm that not all aspects of an ERP implementation can be planned and controlled in advance. They should perform their roles accordingly. Because a paradigm determines what we perceive and how we react to situations, it is important to be aware of the underlying paradigm when performing research or managing an ERP implementation in practice. Therefore research into the complexity of ERP implementation should only be conducted by explicit application of a specific complexity research approach. If we reason from Manson's differentiation, this should be a deterministic or an aggregate complexity research approach. Because the deterministic paradigm predominates in current ERP research, we recommend more research by an aggregate complexity research approach. Therefore, we recommend further research into appropriate control mechanisms by that approach. We also recommend research based on an aggregate complexity research approach to support a better understanding of results from research based on a deterministic complexity research approach. Also, we like to stimulate research which results can support researchers in deciding on the best research complexity approach for specific ERP implementation research. And finally, we consider it useful to look more profound at other theoretical differentiations for complexity research approaches. These differentiations might offer even more insights in how research and practice should treat complexity of ERP implementation.

Figure 0.1 shows an overall view on our research questions and performed steps.

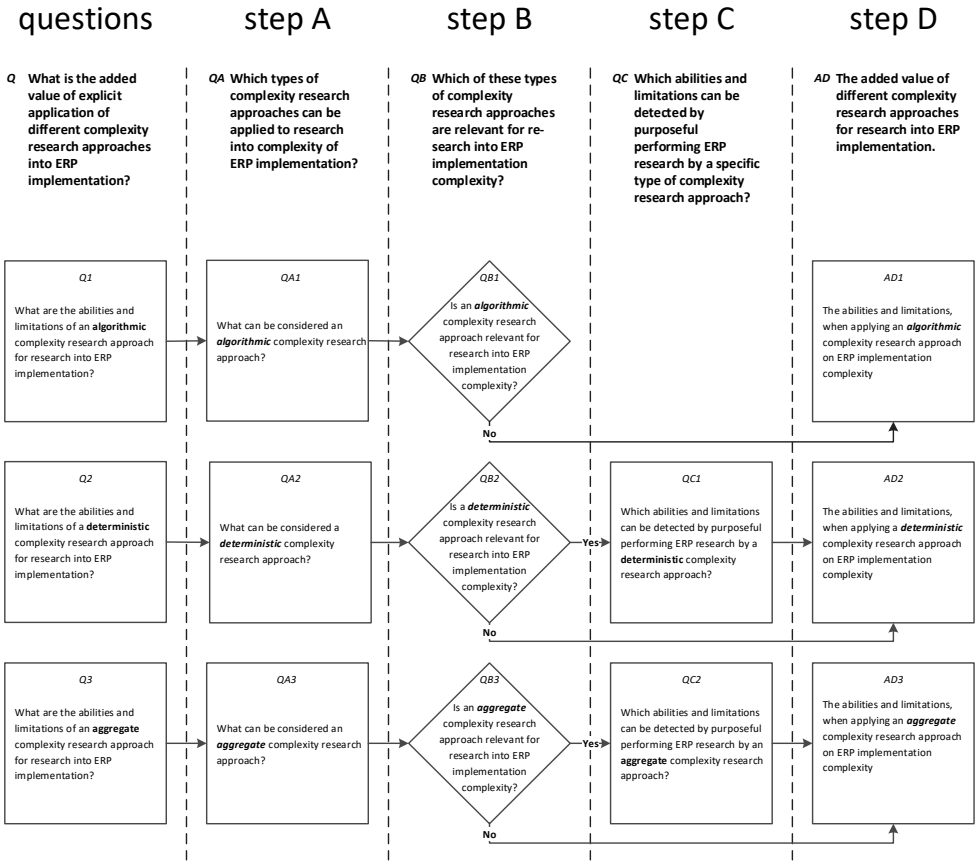


Figure 0.1 Research steps and research questions

Summaries of individual studies

Study 1: complexity impact factors on the integration process of ERP and non-ERP systems

This study shows an expert confirmed initial list of factors that influence the complexity of the integration process of ERP- and non-ERP systems. After a thorough search for complexity factors in the scientific literature, we conducted a survey amongst eight experts of a leading European special steel products company. This company was recently composed out of several independent international companies. The participants confirmed the from literature retrieved list, consisting of five quantitative and 21 qualitative factors. The experts added one extra qualitative factor and scored the importance of all factors. Three quantitative factors, i.e. a technology, business and project factor, scored highest. This initial list of factors can support increasing the complexity awareness in activities such as planning, control and risk management when dealing with integration issues.

Study 2: Sizing ERP Implementation Projects: An Activity-Based Approach

ERP implementation projects affect large parts of an implementing organisation and lead to changes in the way an organisation performs its tasks. The costs needed for the effort to implement these systems are hard to estimate. Research suggests that the size of an ERP project could be a useful measurement for predicting the effort required to complete an ERP implementation project. However, such a metric does not as yet exist. Therefore, research into a set of variables defining the size of an ERP project should be conducted. The authors hypothesise that ERP projects consist of a collection of clustered activities with each their focus on implementation costs and project size. A survey among domain experts confirmed this. This study shows the first step to retrieve these clusters. It shows 21 logical clusters of ERP implementation project activities based on 405 ERP implementation project activities retrieved from literature. Logical clusters of ERP project activities can be used in further research to find variables for defining the size of an ERP project.

Study 3: an expert based taxonomy of ERP implementation activities

ERP implementation projects are complex and expensive. Organisations usually manage this complexity by dividing the project into phases. However, such a division into phases does not seem to enhance the understanding of the underlying processes. This research, therefore, aims at enhancing the understanding of these underlying processes through an expert based taxonomy of implementation activities, independent of time and phasing. We developed this taxonomy by the retrieval of 205 ERP implementation activities from literature, the grouping of these activities by 11 ERP implementation experts and the comparison with study 2. We used Delphi card sorting as the method for grouping which was supported by Websort as a web-based card sort-

ing tool. The proposed taxonomy can serve as a base for further research into ERP implementation projects and can support the management of ERP projects.

Study 4: exploratory research into the existence of unexpected issues in ERP implementations

The goal of this study was to conduct some exploratory research to validate the existence of unexpected behaviour in ERP implementations. We detected this unexpected behaviour by demonstrating unexpected issues which are clearly out-of-scope of an ERP implementation project and which can only be solved outside the project. We needed characteristics to detect these unexpected issues. Therefore we designed a model of how an ERP implementation project handles issues and defined what types of issues are considered to be unexpected. Next, we carefully selected a case study at a large public body. We detected seven unexpected issues by application of our model in this case study. From our case study results, we conclude that it is reasonable to assume that ERP implementations, despite proper preparation and management, can show unexpected behaviour.

SAMENVATTING

Deze samenvatting bestaat uit twee delen:

1. Het eerste deel bevat de samenvatting die de methodologie, resultaten en conclusies van het hoofddoel van ons onderzoek bespreekt.
2. Het tweede deel bevat de afzonderlijke samenvattingen van de deelonderzoeken.

Samenvatting voor het hoofddoel van het onderzoek

Het implementeren van enterprise resource plannings (ERP) systemen (in dit proefschrift verder genoemd ERP-implementatie) kan worden beschouwd als een complexe aangelegenheid. ERP-implementatie wordt als complex beschouwd, omdat het in de meeste gevallen grote delen van een organisatie beïnvloedt. Ook de kosten van software, hardware, onderhoud, maar vooral het implementatieproces zelf zijn voor een organisatie hoog en risicovol. Daarom streeft onderzoek naar ERP-implementaties om de praktijk te voorzien van nuttige inzichten en tools voor een betere beheersing van de kosten en risico's. ERP-implementaties worden al meer dan 20 jaar uitgevoerd. Helaas hebben zowel de wetenschap als praktijk nog onvoldoende middelen, richtlijnen en methoden opgeleverd voor het adequate beheer en de controle van ERP-implementaties. Na 20 jaar beschouwen onderzoek en praktijk ERP-implementatie nog steeds als een complex onderwerp.
Middels ons onderzoek willen we deze complexiteit van ERP-implementatie benaderen vanuit een nieuw perspectief. Een perspectief dat het construct 'complexiteit' zelf als uitgangspunt neemt.
We hebben ERP-implementatiecomplexiteit bestudeerd als een construct in de context van complexiteitsonderzoek en hebben ons afgevraagd hoe de complexiteit van ERP-implementaties het best kan worden bestudeerd.

Onze hoofdonderzoeksvraag is:

Wat is de toegevoegde waarde van expliciete toepassing van verschillende complexiteits-onderzoeksbenaderingen voor ERP-implementatie?

(Origineel: What is the added value of explicit application of different complexity research approaches into ERP implementation?)

We hebben binnen bestaand ERP-implementatieonderzoek een gestructureerde literatuurstudie naar het gebruik van het construct complexiteit uitgevoerd. Hoewel de termen complex en complexiteit vaak gebruikt worden in ERP onderzoeksresultaten, kwamen we vrijwel geen bespreking van het construct complexiteit met betrekking tot ERP-implementaties tegen. Daarom stelden we op basis van complexiteitsliteratuur een definitie van ERP-implementatiecomplexiteit op. Bovendien merkten we in onze gestructureerde literatuurstudie op, dat we geen expliciet of impliciet aangegeven complexiteitsonderzoeksbenaderingen konden vinden. Deze lacune noodzaakte ons om te concluderen dat, om in staat zijn om onze onderzoeksvragen te beantwoorden, we een praktische differentiatie nodig hadden om de verschillende complexiteitsonderzoeksbenaderingen te kunnen detecteren en te onderscheiden. We hadden deze differentiatie ook nodig om de waarde van elke benadering te kunnen bepalen. Daarom kozen we Mansons drie complexiteitsonderzoeksbenaderingen als de basis voor ons onderzoek:

- algoritmische complexiteitsonderzoeksbenadering
- deterministische complexiteitsonderzoeksbenadering
- aggregaat complexiteitsonderzoeksbenadering

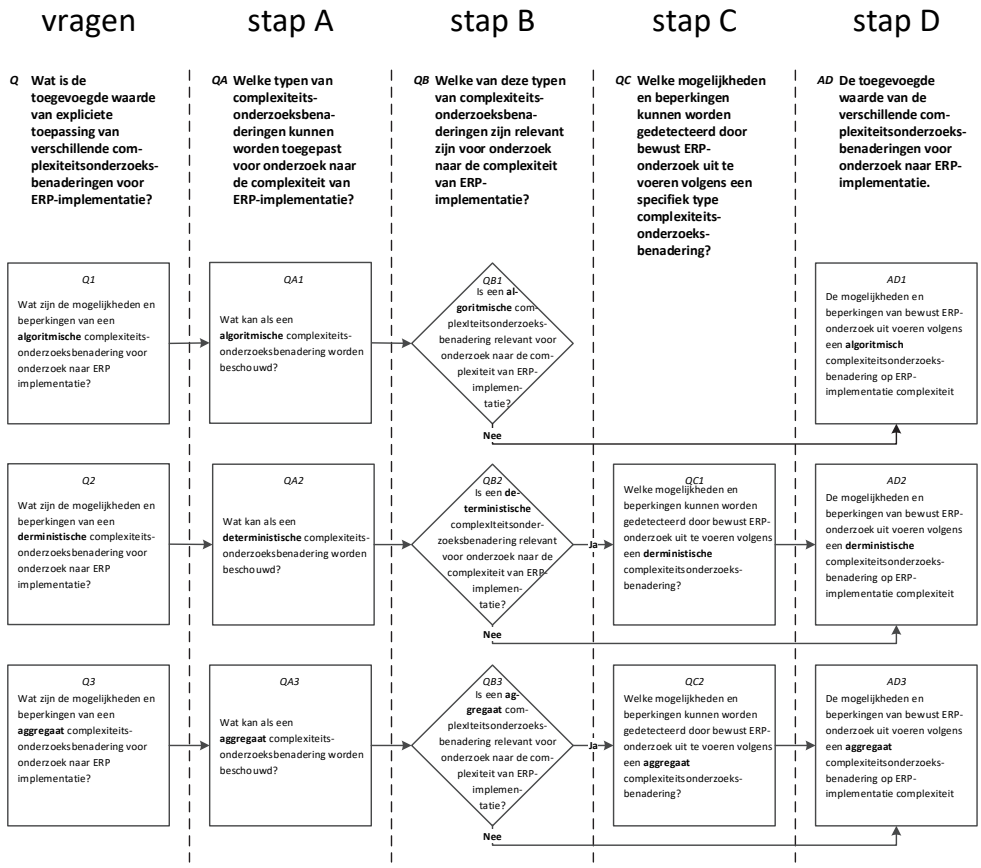
Op basis van de differentiatie van Manson en de resultaten van onze gestructureerde literatuurstudie, concludeerden we dat een deterministische complexiteits-onderzoeksbenadering binnen de heersende stroming van ERP-implementatie-onderzoek de overhand heeft. We hebben in de resultaten van onze literatuurstudie geen algoritmische of aggregaat complexiteitsonderzoeksbenadering gevonden. We hebben de algoritmische complexiteitsonderzoeksbenadering in ons empirisch onderzoek niet verder verkend, omdat we beredeneerden dat deze benadering te beperkt zou zijn als deze gebruikt zou worden als principe voor het begrijpen van de werkelijke complexiteit van ERP-implementatie. Echter voor de deterministische en de aggregaat benaderingen hebben we empirisch onderzoek uitgevoerd om de waarde van beide benaderingen voor onderzoek en praktijk te kunnen bepalen. We hebben drie

deelonderzoeken uitgevoerd volgens een deterministische complexiteitsonderzoeksbenadering en één deelonderzoek volgens een aggregaat complexiteitsonderzoeksbenadering. De resultaten van deze vier deelonderzoeken hebben voldoende inzichten gegeven in de gezochte waarden van de deterministische en aggregaat complexiteitsonderzoeksbenaderingen. Naast deze inzichten leverden deze vier deelonderzoeken ook zinvolle resultaten op als afzonderlijke onderzoeken.
Gebaseerd op de resultaten van deze vier deelonderzoeken concluderen we dat zowel de deterministische als de aggregaat complexiteitsbenadering waardevol lijken, elkaar kunnen aanvullen en dus relevant zijn voor wetenschappelijk onderzoek en voor de praktijk. We concluderen dat het zinvol is om expliciet volgens een aggregaat complexiteitsonderzoeksbenadering meer onderzoek te doen naar ERP-implementatie. Het uitvoeren van meer onderzoek volgens een aggregaat complexiteitsonderzoeksbenadering zou wel eens de resultaten van onderzoeken volgens een deterministische complexiteitsonderzoeksbenadering kunnen aanvullen, leiden tot verschillende gezichtspunten voor ERP-implementaties en nieuwe inzichten verschaffen voor zowel onderzoek als praktijk.

We vinden dat het belangrijk is om zich bewust te blijven van het paradigma waarop een complexiteitsonderzoeksbenadering is gebaseerd. Het managen van een ERP-implementatieproject met in het achterhoofd alleen een deterministisch complexiteitsparadigma, bevat het risico van een vals gevoel van veiligheid, wanneer uitsluitend op factoren en richtlijnen afgeleid van deterministische ERP-implementatie complexiteit onderzoek vertrouwd wordt. Bovendien zouden het topmanagement, leden van het ERP-implementatieproject en andere stakeholders zich volgens het aggregaat paradigma bewust moeten zijn dat niet alle aspecten van een ERP-implementatie op voorhand kunnen worden gepland en beheerst. Ze zouden hun rol dienovereenkomstig moeten uitvoeren. Omdat een paradigma bepaalt wat we waarnemen en hoe we reageren op situaties, is het belangrijk om zich bij het uitvoeren van onderzoek of managing van een ERP-implementatie in de praktijk bewust te zijn van het onderliggende paradigma. Daarom zou ieder onderzoek naar de complexiteit van ERP-implementatie alleen moeten worden uitgevoerd met expliciete toepassing van een specifieke complexiteitsonderzoeksbenadering. Als we redeneren vanuit de differentiatie van Manson, dan zou dit een deterministische of aggregaat complexiteitsonderzoeksbenadering moeten zijn. Omdat het deterministische paradigma de boventoon voert in bestaand ERP onderzoek, bevelen we verder onderzoek volgens een aggregaat complexiteitsonderzoeksbenadering aan. Daarom bevelen we meer onderzoek naar geschikte beheers-mechanismen volgens die benadering aan. Om een beter begrip te krijgen van de resultaten uit wetenschappelijk onderzoek gebaseerd op een deterministische complexiteitsonderzoeksbenadering, bevelen we ook verder onderzoek gebaseerd op een aggregaat complexiteitsonderzoeksbenadering aan. We zouden bovendien onderzoek willen stimuleren, waarvan de resultaten onderzoek naar het beslissen over de beste research complexiteitsbenadering voor specifiek ERP-

implementatie onderzoek zouden kunnen ondersteunen. En tenslotte: we beschouwen het zinvol om diepgaander naar andere theoretische differentiaties van complexiteitsonderzoeksbenaderingen te kijken. Deze differentiaties zouden nog meer inzicht kunnen bieden in hoe onderzoek en praktijk de complexiteit van ERP-implementaties zouden moeten beheersen.

Figuur 0.2 laat een overzicht zien van onze onderzoeksvragen en uitgevoerde stappen



Figuur 0.2 Onderzoeksstappen en onderzoeksvragen

Samenvattingen van de deelonderzoeken

Deelonderzoek 1: complexiteit invloed factoren op het integratieproces van ERP en niet-ERP systemen

Deze studie presenteert een eerste door experts bevestigde lijst van factoren die de complexiteit van het integratieproces van ERP-systemen en niet-ERP-systemen beïnvloeden. Na een uitgebreide zoektocht naar complexiteitsfactoren in de wetenschappelijke literatuur, hebben we een onderzoek uitgevoerd onder acht experts in een toonaangevende Europees speciale staalproducten bedrijf. Dit bedrijf was nog niet lang geleden samengesteld uit verschillende onafhankelijke internationale bedrijven. De experts bevestigden de samengestelde lijst uit de literatuur, bestaande uit vijf kwantitatieve en 21 kwalitatieve factoren. De experts voegden één extra kwalitatieve factor toe en scoorden het belang van alle factoren. Drie kwantitatieve factoren kregen de hoogste score, dat wil zeggen een technologie, een bedrijfs- en een projectfactor. Deze eerste lijst van factoren kan het complexiteitsbewustzijn bij organisaties verhogen in activiteiten zoals planning, controle en risicobeheer bij de behandeling van integratieproblemen.

Deelonderzoek 2: De omvang van ERP-implementatieprojecten: een benadering via activiteiten

ERP-implementatieprojecten zijn van invloed op grote delen van een organisatie en leiden tot veranderingen in de manier waarop een organisatie haar taken uitvoert. De kosten die nodig zijn om deze systemen te implementeren zijn moeilijk te schatten. Onderzoek suggereert dat de omvang van een ERP-project een bruikbare maat zou kunnen zijn voor het voorspellen van de inspanning die nodig is om een ERP-implementatieproject uit te voeren. Echter, een dergelijke maat is er nog niet. Daarom zou onderzoek moeten worden uitgevoerd om een reeks variabelen te vinden die de grootte van een ERP-project zouden kunnen bepalen. De auteurs veronderstellen dat ERP-projecten bestaan uit een verzameling van clusters van activiteiten met ieder hun eigen focus op de kosten van de uitvoering en omvang van de projecten. Dit werd bevestigd in een enquête onder domein experts. Dit deelonderzoek laat de eerste stap zien voor het bepalen van deze clusters. Het toont op basis van 405 uit de literatuur opgehaalde ERP-implementatieprojectactiviteiten 21 logische clusters van ERP-implementatieprojectactiviteiten. Logische clusters van ERP-project activiteiten kunnen in toekomstig onderzoek worden gebruikt om variabelen te vinden voor het bepalen van de grootte van een ERP-project.

Deelonderzoek 3: een op experts gebaseerde taxonomie van ERP-implementatie-activiteiten

ERP-implementatieprojecten zijn complex en duur. Doorgaans beheersen organisaties deze complexiteit door het splitsen van het project in fasen. Echter het opsplitsen van het project in fasen lijkt het begrip van de onderliggende processen niet te verbeteren.

Daarom is dit onderzoek gericht op het verbeteren van het begrip van deze onderliggende processen door middel van een door experts vastgestelde taxonomie op basis van implementatieactiviteiten, welke onafhankelijk zijn van tijd en fasering. Deze taxonomie hebben we ontwikkeld door het uit de literatuur destilleren van 205 ERP-implementatie activiteiten, het groeperen van deze activiteiten met behulp van 11 ERP-implementatie experts en een vergelijking te trekken met deelonderzoek 2. Wij hebben "Delphi card sorting" als methode voor het groeperen gebruikt, ondersteund door Websort als web gebaseerd ondersteunend tool. De voorgestelde taxonomie kan dienen als een basis voor verder onderzoek naar ERP-implementatieprojecten en kan het managen van ERP-projecten ondersteunen.

Deelonderzoek 4: een verkennend onderzoek naar het bestaan van onverwachte problemen in ERP-implementaties

Het doel van dit deelonderzoek was om verkennend onderzoek uit te voeren naar het bestaan van onverwacht gedrag van ERP projecten zelf. We spoorden dit onverwacht bedrag op door onverwachte problemen aan te tonen die duidelijk out-of-scope van een ERP-implementatieproject waren en alleen konden worden opgelost buiten het project. We hadden kenmerken nodig om deze onverwachte problemen op te sporen. Daarom hebben we een conceptueel model gedefinieerd dat aangeeft hoe een ERP-implementatieproject problemen afhandelt en hebben we gedefinieerd welk typen van problemen gezien zouden moeten worden als onverwacht. Vervolgens hebben we zorgvuldig een case bij een grote overheidsinstantie geselecteerd en bestudeerd. We detecteerden zeven onverwachte problemen door toepassing van ons model in deze case. Op basis van de resultaten van ons deelonderzoek concluderen we dat het redelijk is om te veronderstellen dat ERP-implementaties, ondanks een goede voorbereiding en beheer, onverwacht gedrag kunnen vertonen.

CHAPTER 1: INTRODUCTION

1.1 Introduction

Implementing enterprise resource planning (ERP) systems is considered to be a complex matter (Ghosh & Skibniewski, 2010; Grabski, Leech, & Schmidt, 2011; Janssens, Hoeijenbos, & Kusters, 2011). It is considered complex because an implementation in most cases influences large parts of an organisation.
Implementing an ERP (enterprise resource planning) system is also a very expensive affair. The cost of software, hardware, maintenance but especially the implementation process itself are high. The implementation may cause risks to the organisation. Therefore, researchers are interested in the implementation process of ERP systems. Research can provide practice with useful insights and tools for improved management of both costs and risks of this process. Over the years, considerable research has been performed into the implementation process itself. Researchers consider implementing an ERP system to be of a complex nature. For instance, a search on Google Scholar for "Enterprise Resource Planning" and "complex" results in over 50,000 hits. Our research, as described in this thesis, aims at providing more insight into the complexity of this implementation process.
First, we will define in this chapter the notions of "ERP system" and "ERP implementation". Next, we will provide a short overview of the main research themes within ERP implementation. After that, we will motivate the relevance of our research and present an overview of this thesis.

1.2 ERP systems

ERP systems as a concept came into being in the 1970's. However, only in the 1990's ERP systems started to show real benefits for organisations (Dey, Clegg, & Bennett, 2010a). ERP evolved from MRP (Material Requirements Planning) and MRPII (Manufacturing Resource Planning), which mainly focused on optimisation of production processes (Siau, 2004). Later this type of software also supported other business processes within organisations. For instance, it supports financial and human resource management. The name Enterprise Resource Planning was suggested by the Gartner Group (Mabert, Soni, & Venkataramanan, 2003) and has been used since for this type of systems.

An ERP system can be defined as (Klaus, Rosemann, & Gable, 2000):

ERP system
ERP systems are comprehensive, packaged software solutions which seek to integrate the complete range of a business's processes and functions in order to present a holistic view of the business from a single information and IT architecture.

An ERP system is software which consists of different modules like human resources, sales, finance and productions. These modules support business processes of organisations (Nazemi, Tarokh, & Djavanshir, 2012). The most significant benefit of an ERP system is the integration of these business processes (Klaus et al., 2000; Motwani, Subramanian, & Gopalakrishna, 2005). Another significant benefit is the possibility for organisations for replacement of largely fragmented information systems (Ahmad & Pinedo Cuenca, 2013; Boudreau, Robey, Marie-Claude, & Daniel, 1999). Because for the instant availability and lesser cost for development, organisations prefer ERP systems over the in-house development of systems. Also, the overall integration of business processes and transactions are of value to implementing organisations. Unfortunately precisely this integration of these business processes and built-in standardisation of these business processes, also cause problems when an organisation implements an ERP system. Often an ERP system will replace own custom made software. In most cases, implementation of an ERP system in an organisation causes significant changes throughout the organisation. An ERP implementation severely influences how an organisation handles its business (Rosa, Packard, Krupanand, Bilbro, & Hodal, 2013). Therefore the implementation process itself is of complex nature and has to be handled with care.

1.3 ERP implementation

ERP implementations introduce an ERP information system or parts of an ERP information system (one or more modules) into an organisation. Literature clearly describes and defines what an ERP information system embodies. However, we did not retrieve an explicit definition regarding an ERP implementation in our encountered literature. Therefore we considered it for our research necessary to firstly define what we consider an ERP implementation.
In most cases, ERP implementations are discussed and handled as projects, as is indicated by a vast amount of research into ERP projects as a subject (Fadlalla & Amani, 2015; Nazemi, Tarokh, & Djavanshir, 2012; Schlichter & Kraemmergaard, 2010).
Hence we turn to project management as a base for our definition. In project management the building blocks or elements of projects are well-known. A project always consists of activities. These activities need resources (human and non-human) which

perform these activities. The results of these activities are (sub) products. Products which are required by or are of interest to stakeholders. These activities, products and stakeholders are interconnected to each other and can influence each other (Meredith & Mantel Jr, 2011). Besides these project aspects of an ERP implementation, a fundamental aspect is its organisational impact (Grabski et al., 2011; Schniederjans, 2013). In most cases, it changes the way an organisation operates by altering its business processes. Research shows that organizational change (Kwahk, 2006; Wei, Chien, & Wang, 2005), and as a consequence change management, is a very important aspect of an ERP implementation in order to successfully implement an ERP system (Altamony, Al-Salti, Gharaibeh, & Elyas, 2016; Marnewick & Labuschagne, 2005; Ngai, Law, & Wat, 2008; Shaul & Tauber, 2013). Therefore we consider it essential that this organisational change aspect should also be a part of an ERP implementation definition.
We define ERP implementation for our research based on the general elements of project management and the organisational change aspect:

ERP implementation
All activities undertaken, resources needed, (sub)products produced, stakeholders, and their interrelationships to introduce (parts of) an ERP information system in an organisation and the associated necessary organisational changes.

1.4 Research context

There has been considerable research into ERP implementation on a broad range of topics in the last 20 years. For instance, a search in Google Scholar on "Enterprise resource planning implementation" over 1995 to 2015 shows about 76,000 hits. The same search restricted to 2015 shows about 4,200 hits. It is necessary to outline the most prominent topics within ERP implementation research to be able to position our research in scientific context. Therefore we identify in this section what we consider the main research themes for ERP implementation.

Implementations are still over budget, time and below expectations of stakeholders, although practice and research learned substantial lessons on ERP implementation for over 20 years (Davide Aloini, Dulmin, & Mininno, 2012b). Therefore research and practice have put and still put considerable effort in reducing and handling the complexity of ERP implementation. Research tries to contribute through several themes. We will discuss only briefly the main themes we encountered in the research literature because exploring these research themes was not the base or purpose for our research. Various researchers have provided an overview of research within the

ERP field of research. Because we focus on the implementation of ERP systems in our research, we will restrict our discussion to implementation related themes.

Moon (2007) distinguished the following themes for research into ERP implementation:

- General
- Case study
- Critical success factors
- Change management
- Focused stage in the implementation process
- Cultural (national) issues

Schlichter et al. (2010) encountered nine categories of ERP research. Within their category "ERP implementation research" they list the next issues:

- Selection of the ERP system
- Various steps of implementation and related problems
- Critical success factors (CSFs)
- Business process alignment during the implementation (BPR)
- Organizational diffusion

Fadlalla and Amani (2015) developed an objective keyword-based framework. Their framework can be used as an organising tool for ERP research contributions. Table 1.1 shows the core topics they discovered in ERP research literature by this framework.

Table 1.1 Emerged core ERP research topics for the period: 2000-2013 (Fadlalla & Amani, 2015)

Core topic	Justification
Benefits	A key business driver of acquiring an ERP system
BPR	Essential prerequisite for a successful ERP implementation
Case study	The most common method in ERP research
Change management	Necessary for a successful ERP implementation
Company performance	Necessity of measuring the impact of ERP systems on company performance
Competitive advantage	The role of ERP in supporting companies to achieve competitive advantage
CSFs	To successfully implement an ERP system, it is imperative to study and understand the critical success factors of ERP implementation
Customization	Improving the fit between the company business processes and the pre-packaged processes in an ERP system
Consultancy	The significant role of consultancy in the implementation of an ERP
Decision making	The central role of ERP systems as a source of real-time information to enable effective decision-making
Implementation	The strategic nature of ERP systems, their complexity, and their high cost makes their implementation one of the most researched topics
Innovation	The role of the ERP as an enabler of business process innovation The single most important justification for adopting an ERP system Significant knowledge is essential for selecting, implementing, and sustaining an ERP system
Integration	The single most important justification for adopting an ERP system
Knowledge management	Significant knowledge is essential for selecting, implementing, and sustaining an ERP system
IS	ERP is the most comprehensive business information system
IT	ERP systems have major information technology ramifications
Manufacturing	Certainly the father of ERP systems and remains to be the biggest domain for ERP implementations
Modeling	Essential for leveraging the vast ERP data and capabilities
Organizational change	Is a certainty if ERP systems are to be successfully implemented
Project management	The complexity of ERP projects necessitates properly managing them as such
Resource management	A key business justification for implementing an ERP system
Risk management	The complexity of implementing an ERP system necessitates the importance of managing different types of risk
SCM	The need for going beyond intra-enterprise integration into inter-enterprise integration

Core topic	Justification
Selection	ERP organisational fit is a key success factor, and selection is essential for ensuring such fit
SMEs	The newest, and possibly the hottest, battle ground for ERP systems

Based on the implementation themes from Moon (2007), the implementation issues from Schlichter et al. (2010), and the applicable topics for ERP implementation as intended in our research from Fadlalla and Amani (2015), we summarize these subjects in the following overview:

Research into ERP implementation phases
There are several directions for finding solutions for handling the complexity of ERP implementation. An ERP implementation is in general considered a project. Projects can be divided into several phases or stages. By dividing a project into phases, an organisation can focus on the goals and outcomes of every phase, and therefore simplify the process. Researchers performed considerable research on stages or phases of ERP implementation projects. The determination of which phases a project should be constructed and what activities should be carried out during these phases are relevant guidelines for practice. The number of phases distinguished for ERP implementation differs from three to six (Shaul & Tauber, 2013). According to Shaul and Tauber (2013) the phasing of Markus and Tanis (2000) - planning, implementation, stabilisation of the ERP system into normal operation, and enhancement - is popular and frequently cited.

Research into Critical Success Factors (CSFs) for ERP implementation
Research has also been focusing on determining critical success factors for reduction of complexity of ERP implementation. If an ERP implementation satisfies these factors, the chance for a successful implementation and thus reducing implementation complexity is enhanced. Over the years a considerable number of papers have been published which aim to present the most complete and accurate CSF list for ERP implementation (Shaul & Tauber, 2013).

Research into risks of ERP implementation
Collecting potential risks for ERP implementation can be seen in the same order of magnitude as CSFs. Being aware of possible risks is also a mechanism to manage the complexity of ERP implementation. An organisation can establish mechanisms in advance by which an ERP implementation can avoid risks, or can handle the consequences of possible incidents. As with CSFs, also risks have received considerable attention (Davide Aloini et al., 2012b).

Research into prediction methods for ERP implementation
Another important theme we retrieved in the research literature is the premise that prediction models can be designed by learning from past ERP implementations. Researchers can design models by which practice can predict costs, time and even success. For instance, some researchers define the "size of an ERP project" (Arb, 1997; Francalanci, Iar, & Lanci, 2001; Stensrud, 2001) . Others combine methods and variables from software development into equations for prediction of cost and time (Daneva, 2010; Hansen, 2006; Magnusson, Nilsson, & Carlsson, 2004).

Research into business process redesign for ERP implementation
In most cases, organisations also have to change more or less their business processes as a consequence of implementing ERP systems. A perfect fit between system and processes almost never exists. Sometimes even the ERP system is used as a lever for change (Koch, 2001). Researchers design guidelines or frameworks for proper BPR during ERP implementation (Esteves, Pastor, & Casanovas, 2002; Y. Hwang & Leitch, 2005).

1.5 Research motive

In 1.3 we have discussed that organisations commit considerable effort to managing ERP implementations. However, according to Amid et al. (2012) "It is said that about 70% of ERP implementations fail to deliver anticipated benefits and three-quarters of these projects are unsuccessful. These projects are, on average, 178% over budget, took 2.5 times longer than intended and delivered only 30% of promised benefit". Apparently still little progress has been demonstrated in practice despite all this research. For instance, a general search on the internet with the keywords "ERP failure" reveals numerous sites which discuss collections of cases (for example the ten biggest failures). In these cases, a discussed ERP implementation is described as a "disaster", a "failure", a "disappointment" et cetera. Descriptions of these cases show that these happened not only in the remote past but that many occurred recently. For instance, the Department of Defense in the Netherlands started the ERP implementation project SPEER in 2005 and planned the project to complete before 2007 (Burg van der, Vos, Schimmel, & Poecke van, 2013). It formally ended in 2013 with only 80% of the intended functionality. Also, only 50% of legacy systems migrated to the ERP system. The original budget was about 185 million euro and the actual cost until 2013 summed up to about 900 million euros.
ERP implementations are considered to be very complex projects (Ghosh & Skibniewski, 2010; Grabski et al., 2011; Janssens et al., 2011). It is hardly surprising that they are considered complex projects. An ERP implementation project not only introduces new technology in an organisation but in general also causes organisational changes. The complexity of ERP projects, resulting from the interaction

of technology and organisational changes, makes them hard to manage. Managing this complexity has not yet been solved within the research themes we discussed in the previous section.
As will be demonstrated in Chapter 3, in research the constructs of 'complexity' and 'ERP implementation' are closely related. ERP implementation is in literature frequently referred to as 'complex'. However, it is remarkable that researchers usually omit the definition of this construct of 'complexity'. We will demonstrate this omission in Chapter 3. As far as we are aware, research has not explicitly reflected on the construct of complexity for ERP implementation. Our research will investigate a new point of view on ERP implementation research. We will take a step back and study complexity in ERP implementation as a construct from a higher level of abstraction. We assume that a better understanding of the construct of complexity specific for ERP implementation can enhance research and practice in this field by adding a new viewpoint.
Therefore, as a contribution to science and practice, it seems relevant to further explore the very construct of complexity in relation to ERP implementation.

1.6 Thesis structure

Chapter 2
In Chapter 2 our research approach will be discussed.
First, we will explain the main research questions and present an overview of the research steps we took to answer these research questions. We will motivate each step in detail and briefly discuss purpose and contribution of every individually performed study to our main research questions.

Chapter 3
First, we will present method, results and discussion of a structured literature review on ERP implementation and complexity. Next, we will discuss the general construct of complexity and derive a definition for ERP implementation complexity. Then, we will motivate and present Manson's research approaches into complexity as a basis for our further research. Finally, we will argue which of these complexity research approaches are relevant to our research.

Chapter 4
In Chapter 4 we will explore the value of a deterministic complexity research approach by presenting goals, methods and results from three studies we performed following a deterministic complexity research approach. We will present results and conclusions of these individual studies. At the end of the chapter, we will also present conclusions on a higher level of abstraction to identify abilities and limitations of this research approach.

Chapter 5
Analogues to Chapter 4, we will explore the value of an aggregate complexity research approach by presenting goals, methods and results from a study we performed following an aggregate complexity research approach. Also for this study, we will present individual results and conclusions, but also conclusions which we can draw on a higher level of abstraction to identify the abilities and limitations of this research approach.

Chapter 6
Based on the results and conclusions of Chapters 4 and 5, we will in Chapter 6 answer our main research question and reflect upon our research and provide recommendations for further research.

CHAPTER 2: RESEARCH APPROACH

2.1 Introduction

In this chapter, we will explain our research objectives and approach.
First, we will explain the main research questions and present an overview of the research steps we took to answer our research questions. Next, we will motivate each step in detail and shortly discuss purpose and contribution of every performed individual study to our research questions. Figure 2.1 shows the research structure as guidance to support the reader.

2.2 Research objective and design

In this thesis our general research goal is:

Enhancing understanding of the complexity of ERP implementation.

The purpose of our research is to determine whether a further understanding of the construct of complexity in the field of ERP implementation will enable better research in this area.

We believe that we can make a valuable contribution by taking as a perspective complexity itself as a characteristic of ERP implementation. Therefore we studied ERP implementation in the context of complexity research. The intricacy of the construct of complexity itself is a vital issue as will be discussed in Chapter 3. Complexity is often considered a black box or container type of construct. However, we expect that more insight on complexity can be achieved by dividing the construct of complexity into more meaningful elements. We selected for this differentiation a practical approach provided by Manson (2001). In his review paper on complexity: "Simplifying complexity: a review of complexity theory", he discussed three complexity paradigms: algorithmic, deterministic and aggregate, as the basis of three complexity research approaches. In Chapter 3 we will extensively discuss and explain the construct of complexity and the paradigms behind these three complexity approaches. Manson's

research approaches will serve as the basis for our research and thesis into the complexity of ERP implementation.
We used Manson's three complexity research approaches to be able to study ERP implementation complexity in the context of complexity research. We explored every complexity research approach to determine its abilities and limitations in the context of research into the complexity of ERP implementation.
We started our research by determining whether all three of Manson's approaches are relevant for ERP implementation research. Is the complexity of ERP implementation of such a nature that all three research approaches are relevant for research and practice? We also investigated whether existing ERP research utilises these complexity research approaches explicitly. If we determined an approach to be theoretically relevant, the next step was to conduct research in practice. By doing so, we explored the abilities and limitations of each of Manson's complexity research approaches in the context of ERP implementation. By these results, we could express an opinion on the value of explicitly using complexity approaches in ERP implementation research. Complexity approaches which can enhance the understanding of the complexity of ERP implementation.

2.3 Research questions

As main research question we aimed at answering:

Q	What is the added value of explicit application of different complexity research approaches into ERP implementation?

In figure 2.1 we present the structure of our research (sub) questions and performed steps to provide an answer to this main research question.

We adopted the three approaches on complexity by Manson, which in Chapter 3 will be further motivated and explained. We attempted to determine for every complexity research approach; whether it applies to ERP implementation and whether application of that approach could enhance understanding of the complexity of ERP implementation.

In our research, we aimed at gaining insight into the abilities and limitations of every complexity approach for research and practice of ERP implementation by asking ourselves the next sub-questions to our main research question.

Q	What is the added value of explicit application of different complexity research approaches into ERP implementation?	
	Q1	What are the abilities and limitations of an ***algorithmic*** complexity research approach for research into ERP implementation?
	Q2	What are the abilities and limitations of a ***deterministic*** complexity research approach for research into ERP implementation?
	Q3	What are the abilities and limitations of an ***aggregate*** complexity research approach for research into ERP implementation?

2.4 Research methodology

2.4.1 Introduction

We will discuss our research methodology on two levels of abstraction. First, on the overall methodology level, we will explain the various steps we took in answering our main research question. Figure 2.1 shows these steps labelled A, B and C. Second; we will shortly discuss our performed research methodology for step C. In step C we conducted four studies. We will provide a detailed discussion of the methodology for every study in step C in the appropriate sections in our thesis. Also, we will discuss validity and reliability of the used methods in these corresponding sections.
Figure 2.1 shows an overall view on our research questions and performed steps. We will use this figure in a simplified version in each chapter, to show which step or part of a step we will discuss in that chapter or subsection.

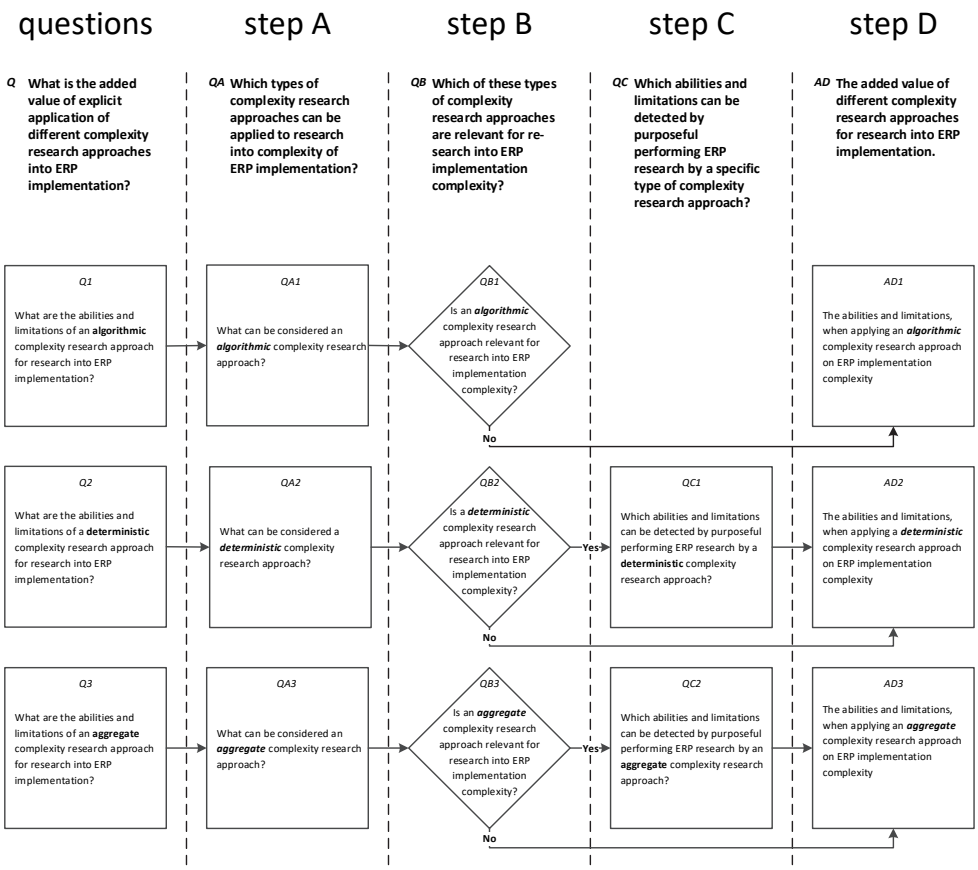


Figure 2.1 Research steps and research questions

2.4.2 Overall methodology

As already mentioned before, we consider complexity an important characteristic of ERP implementation. ERP implementation research is performed to understand this complexity better and aims at providing practice with better tools for handling ERP implementations. We also intend to contribute through our research to a better understanding of the complexity of ERP implementation. However, we intend to approach this from, as far as we are aware, a novel perspective. This novel perspective takes the construct of complexity itself as a starting point. We intend to approach the construct of the complexity of ERP implementation itself in a systematic manner. Though, what is a suitable systematic manner for researching this construct of complexity? We needed a suitable differentiation for analysis of this construct. As will be explained

later, science discusses complexity in numerous ways and various disciplines. These disciplines range from fundamental philosophy to physical sciences and approach research into complexity in different ways. In our search for better understanding the construct of complexity using various complexity research approaches, we encountered Manson's differentiation. Manson aimed at providing an overview of the various approaches independent of a discipline. We consider this overview a clear and also practical basis for our research. Therefore, we chose to adopt the three complexity approaches as discussed by Manson (2001) as foundation for our research:

- algorithmic complexity research approach
- deterministic complexity research approach
- aggregate complexity research approach

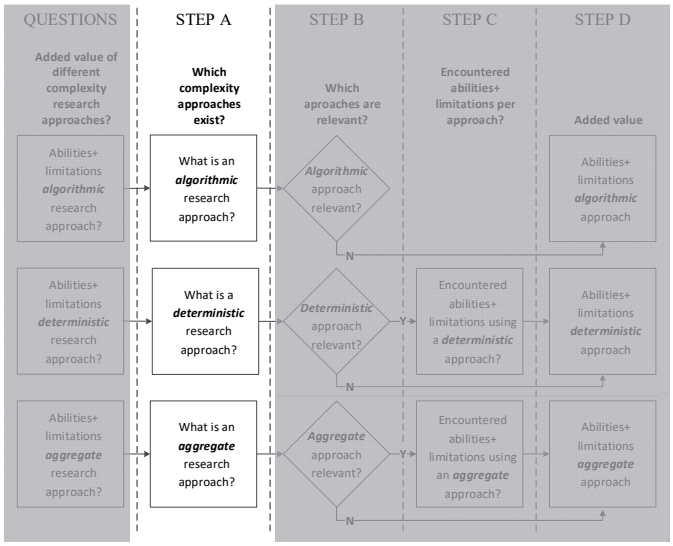
We will further explain Manson's complexity research approaches in Chapter 3. To be able to answer our main research question: "What is the added value of explicit application of different complexity research approaches into ERP implementation?", we designed our research around these three complexity approaches.

As a consequence of taking a new angle in research on a specific subject, first studies in a new direction will be of explorative nature. Our used methodology reflects this explorative nature. An overall qualitative research design seemed most suitable, as we intend to achieve getting a first impression of the value of the three complexity research approaches for ERP implementation.

We aimed at using the best-fit research methods for the research questions which we answered by the results of our empirical studies. Therefore we used for every study different research methods.

Next, we will discuss every step in detail.

2.4.3 Step A

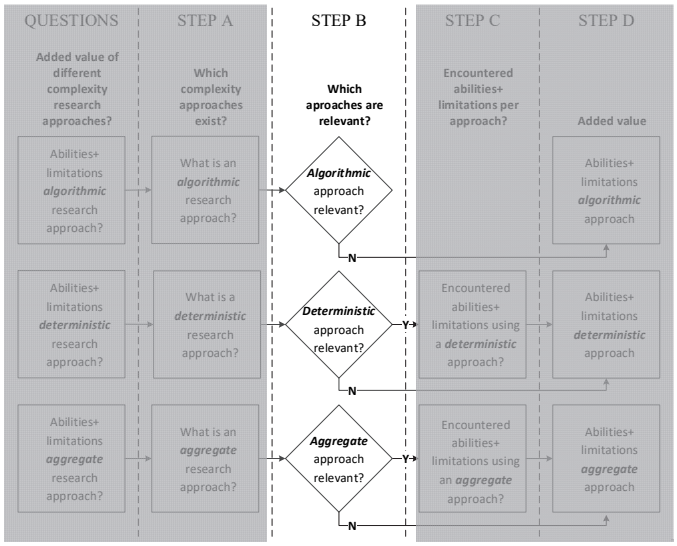


As a first step, we explored the construct of complexity itself and its use in ERP research. Next, we explored the main characteristics of the three complexity research approaches by Manson.

QA	Which types of complexity research approaches can be applied to research into complexity of ERP implementation?	
	QA1	What can be considered an ***algorithmic*** complexity research approach?
	QA2	What can be considered a ***deterministic*** complexity research approach?
	QA3	What can be considered an ***aggregate*** complexity research approach?

We performed a structured literature study after the use of the construct of complexity in existing ERP implementation research. We concluded that only two of our detected papers discussed complexity for ERP implementation in a more profound manner. We considered that only describing or defining complexity is insufficient for performing structured research into the complexity of ERP implementation. Therefore, we adopted Manson's three complexity research approaches for further structuring our research into relevant complexity approaches for ERP implementation research.

2.4.4 Step B



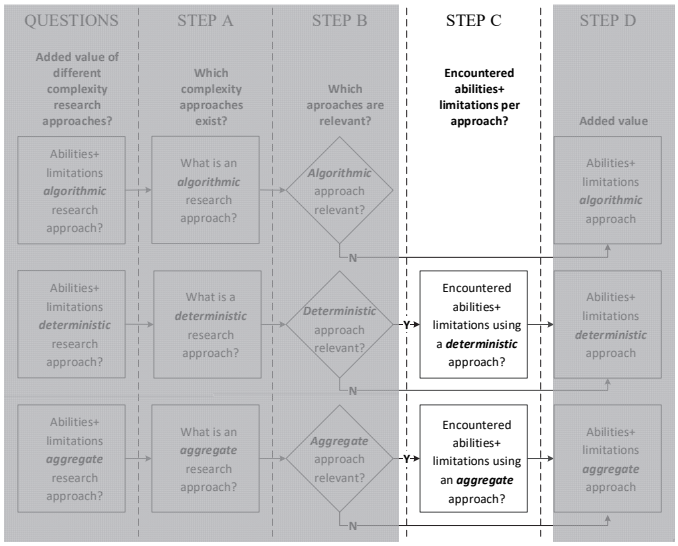
We reasoned whether every complexity research approach makes sense when performing research into the complexity of ERP implementation. We based this reasoning on the characteristics of the complexity research approaches and characteristics of ERP implementation.

QB	Which of these types of complexity research approaches are relevant for research into ERP implementation complexity?	
	QB1	Is an ***algorithmic*** complexity research approach relevant for research into ERP implementation complexity?
	QB2	Is a ***deterministic*** complexity research approach relevant for research into ERP implementation complexity?
	QB3	Is an ***aggregate*** complexity research approach relevant for research into ERP implementation complexity?

Manson divides complexity research approaches into three types. However, we investigated whether all three complexity research approaches are from a theoretical point of view relevant for ERP implementation research and practice. If a complexity research approach from a theoretical point of view would not be relevant, then a further exploration of the values of that approach would not make sense. We formed an opinion on their relevance by comparing the characteristics of the three complexity approaches with ERP implementation characteristics. We concluded that we consider the algorithmic complexity research approach for our research of too little value and therefore not relevant. For that reason, we did not explore the algorithmic complexity approach with regard to ERP implementation in the next steps any further. However,

the deterministic and aggregate research approaches we considered of value and also relevant for further exploration in our next steps.

2.4.5 Step C



We explored the deterministic and aggregate complexity research approaches by performing empirical research based on one of these two. By conducting empirical research, we aimed at better understanding the abilities and limitations of each complexity approach for ERP implementation.

QC	Which abilities and limitations can be detected by purposeful performing ERP research by a specific type of complexity research approach?	
	QC1	Which abilities and limitations can be detected by purposeful performing ERP research by a ***deterministic*** complexity research approach?
	QC2	Which abilities and limitations can be detected by purposeful performing ERP research by an ***aggregate*** complexity research approach?

We performed four studies for the relevant complexity research approaches; that is to say, the deterministic and aggregate complexity approaches.
In this step, we aimed at adding relevant research to mainstream ERP research. Besides adding to mainstream research, we also intended to gain a better insight into the abilities and limitations of the two approaches. That is to say, reflecting on these studies on a higher level of abstraction.
In step B we concluded that both approaches are relevant for ERP implementation research and practice. However, in step B we also assumed by the results of our literature search, that ERP implementation research by an aggregate complexity approach is rare. Therefore in step C we explored the aggregate complexity research approach by explicitly performing a study by this approach.

For the deterministic and aggregate complexity approaches, we will briefly discuss performed studies and used methodologies in these four studies. A comprehensive discussion of the research goals, methodology, results and conclusions will be discussed in the respective chapters.

*Step C: performed research by a **deterministic** complexity research approach*

We performed three studies by applying a deterministic complexity research approach.

Study 1: Which factors influence the complexity of the integration process of ERP systems and non-ERP systems?

In mainstream ERP research, research into critical success factors, risk factors and other influencing factors is extensive. Therefore we intended to expand this knowledge about factors which affect the complexity. We expanded the knowledge by detecting factors which influence the complexity of the integration process of ERP systems and non-ERP systems. Hence Study 1 aimed at compiling a list of factors which influence the complexity of the integration process of ERP systems and non-ERP systems. This list of factors can be useful for better judgment of the complexity of this integration process.

Methodology Study 1:
We performed a thorough search for complexity factors in the scientific literature. After which we selected eight experts from a company which was recently composed out of independent international companies. These experts were surveyed about relevancy and completeness of the identified complexity factors by a multi-round survey approach through e-mail. The survey consisted of predefined questions with predefined answers and in round one an additional open question about supplementary factors.

Studies 2 and 3: Which coherent groups of activities commonly exist in ERP implementation?

Characteristics of ERP implementation project phases are in mainstream ERP research discussed and studied. Researchers propose and study different phasings. However, we noticed that a comprehensive collection of ERP activities independent of the different phasings on a deeper level does not exist in ERP research. A comprehensive collection of ERP implementation activities seemed relevant to us, as this might support planning and managing of an ERP implementation project. Therefore, Study 2 and 3 intended to gain better insight into the complexity of ERP implementation by studying which activities ERP projects usually perform. Also, our study intended to view activities independent of a phase of a project and determine which activities are closely related. We researched which activity clusters usually exist within ERP projects. Knowledge of these activity clusters can serve as a taxonomy model which can

support planning and managing of ERP implementations and therefore reduce handling of the complexity of ERP implementation.

Methodology studies 2 and 3
First, these ERP implementation activities had to be retrieved to enable grouping of ERP implementation activities to form a taxonomy. A collection of activities concerning the implementation of ERP systems has been methodical extracted from the literature since we did not encounter a comprehensive collection of these activities in literature.
Second, we had to refine these retrieved activities. Because these activities can appear in literature as synonyms, homonyms and can have different wording. For example 'training', 'education', 'user training', 'training of users' or 'train the users' all refer to the same activity. This refining had to be done in a controlled manner and could only be done by human judgment.
Third, we had to group these refined activities into meaningful collections. Experts were chosen as a source to define these collections. In Study 2 a metaplan technique was used and in Study 3 experts used a card sorting method for this purpose.

*Step C: performed research by an **aggregate** complexity research approach*

We performed one study by applying an aggregate complexity research approach.

Study 4: Can typical aggregate complexity behaviour of ERP implementation be demonstrated in practice?

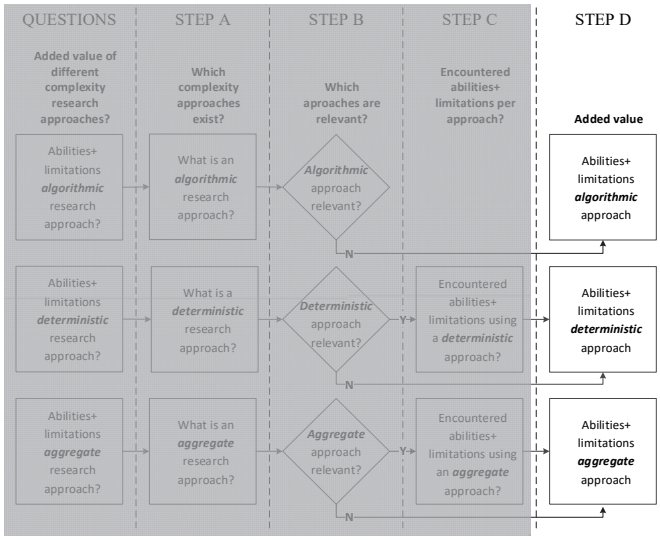
By this study, we explored whether application of an aggregate complexity research approach in research indeed can reveal unexpected behaviour in ERP implementation projects. We consider the unexpected behaviour of projects as unforeseen and unanticipated problems. Problems for which the design of the project was not adequate, although the case organisation professionally designed and managed the project. Also in this study, we intended to get a first impression of whether it is useful to apply an aggregate complexity research approach explicitly.

Methodology Study 4
We performed a comprehensive case study of an appropriate complex ERP implementation. We performed this study to be able to answer whether we could demonstrate typical aggregate complexity behaviour of ERP implementation in practice. We selected as our case an appropriate professionally designed and managed complex ERP implementation project. We retrieved unexpected problems by document analyses and interviews. After that, we determined by the retrieved information and affirmation of the case organisation, whether these problems were indeed unexpected. Finally, we received an acknowledgement from the case organisation that the results

and aggregate complexity view from the aggregate complexity approach provided them with a new view on how to manage complex projects.

From the results of our case study, we concluded that application of the aggregate complexity research approach indeed did reveal unexpected behaviour in this appropriate case. Also, we received positive feedback from the case organisation that this approach might support them in managing future ERP implementations.

2.4.6 Step D



Finally, we could provide answers to our main research questions.
With the results and conclusions from the research questions Q1, Q2 and Q3 we aimed at answering our main research question Q. We were able to discuss the value of each of each complexity research approach for ERP implementation.

AD	The added value of different complexity research approaches for research into ERP implementation.	
	AD1	The abilities and limitations, when applying an ***algorithmic*** complexity research approach on ERP implementation complexity
	AD2	The abilities and limitations, when applying a ***deterministic*** complexity research approach on ERP implementation complexity
	AD3	The abilities and limitations, when applying an ***aggregate*** complexity research approach on ERP implementation complexity?

We concluded for every individual complexity research approach from Manson (2001), its abilities and limitations for research and practice. We concluded this by interpreting our experiences and results of our performed studies.
Finally, we concluded and discussed the value of using different complexity approaches in research and practice by comparing our conclusions from the three complexity research approaches and reasoning based upon these conclusions. Also based

on these overall conclusions, we reasoned what recommendations for further research might be applicable.

2.5 Conclusions

In this chapter, we explained the main purpose of our research. We showed our research questions for every step we took and discussed briefly the results of every step we performed to be able to answer these questions.
In the next chapter, we will first discuss general complexity topics. Then we will discuss step A, the first step in our research.

CHAPTER 3: COMPLEXITY AND ERP IMPLEMENTATIONS

3.1 Introduction

In this chapter, first, we will start with a general discussion of the use of the construct of complexity in ERP research. As the construct of complexity plays a major part in our research, we consider it important to be aware of how current ERP research discusses complexity in relation to ERP implementation. We will present methods and results of a structured literature review on the use of the construct of complexity in ERP research.
Second, we will discuss the construct of complexity in general, to define what we consider ERP implementation complexity in our research.
Third, we will present and explain Manson's differentiation of complexity research approaches as step A in our research. Manson's differentiation is the base for our research.
Fourth, we will discuss which of Manson's complexity approaches we considered relevant for further exploration for our research into ERP implementation complexity. This discussion is step B in our research.

3.2 Complexity as a construct in ERP research: exploring literature

3.2.1 Introduction

Our general research goal is to enhance understanding of the complexity of ERP implementation. Therefore, it is worthwhile to explore the view on complexity within existing ERP research first. We explored whether research has been performed into the complexity of ERP implementation with the construct of complexity explicitly as a base for the design of that research. A first query in scholar.google.com (February 2015) with the terms "Enterprise Resource Planning", "complex" and "complexity" (search 1 in appendix 3.1) resulted in 52,100 hits. Apparently, these terms are often mentioned connected. We performed a more focused and methodical search to retrieve a better understanding of how research perceives the construct of complexity in the area of ERP implementation. We aimed at understanding how research discusses the complexity of ERP implementation and whether the construct of complexity forms the base for the design of ERP research explicitly. For instance: is it discussed in a me-

thodical manner or merely mentioned as a characteristic? Moreover, if methodical discussed, how is complexity approached in that research?
Therefore this section will first discuss method and results of a thorough literature scan within mainstream ERP implementation research, for the purpose of retrieving substantial scientific work on the complexity of ERP implementation.

3.2.2 Literature search strategy

As research often mentions complexity (search 1: 52,100 hits, see appendix 3.1), we focused on a systematic search in a collection of databases (see for this list of databases table 3.1 and appendix 3.2). In these databases, we were able to search into specific areas of papers, like abstract, title, keywords and main text.
Additionally, we searched in **scholar.google.com**, although scholar.google has less functionality for focused searching.

Table 3.1 Searched database with focused search functionality

1.	Academic Search Elite (EBSCO)
2.	ACM Digital Library
3.	ACS Publications
4.	Business Source Premier (EBSCO)
5.	Cambridge University press
6.	Catalogue UM
7.	DOAJ - Directory of Open Access Journals
8.	EBSCO Host
9.	E-Journals (EBSCO)
10.	Emerald [management plus]
11.	ERIC (EBSCO)
12.	Google Scholar / Google Wetenschap
13.	GreenFILE (EBSCO)
14.	HeinOnline
15.	IEEE Digital Library
16.	JSTOR Business, Biological, Mathematics & Statistics Collection
17.	Kluwer Navigator
18.	Lecture Notes in Computer Science
19.	Legal Intelligence
20.	Library, Information Science & Technology Abstracts - LISTA (EBSCO)
21.	NARCIS - the Gateway to Dutch Scientific Information
22.	Nature : international weekly journal of science
23.	OpMaat Premium
24.	Overheid.nl
25.	Oxford Journals
26.	PiCarta (NCC + OLC)
27.	PsycPapers (EBSCO)

28.	Psychology and Behavioral Sciences Collection (EBSCO)
29.	PsycINFO (EBSCO)
30.	PubMed
31.	Regional Business News (EBSCO)
32.	SAGE Journals Online
33.	Science
34.	ScienceDirect (Elsevier)
35.	SpringerLink
36.	Taylor & Francis Group
37.	Web of Science
38.	Wiley Online Library

In our literature search, we were not interested in papers which only mention the word complex somewhere in that paper. We intended to retrieve papers which explicitly use the construct of complexity for research into ERP implementation.
We formulated restricting assumptions about these papers in advance, to be able to efficiently and effectively retrieve these papers. We could translate these restrictions directly to our queries for the databases in table 3.1. For scholar.google we tried to comply with these assumptions as much as possible, as will be discussed in the next section.

We formulated the next assumptions about papers we were interested in:

1. **A paper mainly discussing ERP will always have the string "Enterprise Resource Planning" in its abstract.**

It is reasonable to assume that if enterprise resource planning is the main subject of a research paper, the string "Enterprise Resource Planning" will at least occur in the abstract of the paper.

2. **A paper discussing ERP and complexity can have the strings "complex" or "complexity" in the title and/or abstract, but will always have the strings "complex" or "complexity" in its full-text.**

If complexity is an important topic in the paper, it certainly will be discussed in the paper and therefore should at least occur in the full-text of the paper. Also, it can also occur in the title and abstract of the paper, which even will increase the probability that the topic "complexity" is important in this paper.

3. **A paper discussing ERP, complexity and the construct of complexity itself, probably will contain one or more of the next strings/combinations in the full-text:**

   - "define complex"
   - "define complexity"
   - "definition of complex"
   - "definition of complexity"
   - "what is complex"
   - "what is complexity"
   - "complexity theory"
   - "complex project"

A paper should explain what is meant by the term complexity since we are searching for discussion of complexity in ERP implementation. Therefore, these strings or combinations of these strings probably will appear in its full-text. It is uncommon that an important term in a paper (like "complexity") will be defined in its title or abstract.

In our search strategy, we combined these three assumptions.

3.2.3 Literature search strategy and results

Appendix 3.1 shows our performed database queries by search strings and the number of hits.

First, we searched for papers which contained "Enterprise Resource Planning" and the words "complex" or "complexity" in their titles and retrieved two hits (search 2 in appendix 3.1) for further examination.

Next, we searched for papers containing "Enterprise Resource Planning", and the words "complex" or "complexity" in their abstracts hits (search 3 in appendix 3.1). We retrieved 274 papers. We efficiently needed to determine whether we required evaluating every paper in its entirety. Therefore, we reviewed every abstract of these hits and decided whether a paper was likely to deal in-depth with the complexity or ERP implementation.
Some papers even used the term complexity in the abstract in no relation to ERP. Finally, we selected 73 papers from these 274 hits for a full examination.

After that, we searched for papers containing "Enterprise Resource Planning" in their abstracts but not the words "complex" or "complexity" (to exclude the results of search 3), search 4 in appendix 3.1. We retrieved 26 hits for further examination.

The searches so far provided us with 101 unique papers (duplicates were removed) with high probability for discussing the meaning of complexity of ERP implementation (appendix 3.3).

As scholar.google.com does not have the functionality to search focused with all the three designed restrictions, we performed searches in scholar.google.com after "Enterprise Resource Planning" and "complex" and "complexity". A general search retrieved 51,500 hits. Therefore we limited the results with search 5 and 6 to several keywords only in the abstract or title. Also, we performed searches 7 through 25 which included authors on complexity, a list derived from Hertogh and Westerveld (2009). Appendix 3.1 shows all searches. We retrieved 17 additional papers (duplicates were removed) from 133 hits.

We retrieved in total 117 relevant papers from the databases listed in table 3.1 and from scholar.google.com. Appendix 3.3 shows these papers.

In the next step in our literature search, we reviewed the full content of each paper and determined if and how a paper discussed the complexity of ERP implementation. To be able to do that, we searched with a PDF reader's search function for the string "complex" (which retrieves instances of the word complex as well as complexity) in every paper's PDF-file. We interpreted how and what that paper discussed about complexity about ERP implementation.
We noticed that the use of the term complexity was not always related to ERP implementation. For instance "complex formula", "complex government rules" et cetera. Sometimes even the term complexity itself was not mentioned in the main text, but only in a reference.
From these 117 papers only 67 related the term complexity directly ERP implementation.
Some papers only mentioned complexity theory, without any further explanation or use in their research. However, some papers defined the term complexity.

Table 3.2 shows encountered statements and characteristics of all 117 papers. Table 3.2 illustrates the number of occurrences of the statements and the number of occurrences of the ratings of the characteristics. Of course, a paper could score on more than one statement and characteristic.

Table 3.2 Paper scores on statements and characteristics over 117 reviewed papers

	Scores
Statements	
ERP systems are complex	50
ERP projects are complex	29
ERP implementations are complex	35
ERP projects have complex environments	15
Characteristics	
Occurring of the term "complexity" related to ERP implementations	66
Occurring of the term "complexity" unrelated to ERP implementations	89
Term complexity not found in main paper text	7
Complexity theory is mentioned	7
The term complexity is defined	5

In appendix 3.4 all 117 retrieved papers, the encountered statements and assigned characteristics are shown.

From table 3.2 it can be concluded that the majority of the 117 reviewed papers uses the word complex or complexity related to ERP, or even related to non-ERP subjects, but does not define its meaning. Also, seven papers refer in their text to complexity theory. However, except the five papers who also defined the term complexity, the reviewed papers did not explain or use these theories. Only five papers discuss the term complexity itself related to ERP implementation as intended in this thesis.

Therefore we evaluated these five papers in detail with regard to complexity definition and ERP. We intended to determine how these papers approached complexity in that ERP research, to be able to detect how they treated complexity in their research.

1. Fontana, R. M., & Neto, A. I. (2009). ERP systems implementation in complex organizations. JISTEM - Journal of Information Systems and Technology Management (Online), 6(1), 61-92.
Fontana and Neto performed an exploratory study into ERP implementation. Their research question was: "What are the relevant aspects in the process of organizational changes generated by the implementation of ERP systems?". The general objective of their study was to analyse the implementation of ERP systems based on a proposal for a model of organisational change. They proposed a complex organisational reference model to contribute understanding of the implementation process.
They consider organisations as complex systems. They refer to McCarthy (2000) for a definition of manufacturing organisations as complex adaptive systems by "they consist of an integrated assembly of interacting elements, designed to carry out cooperatively a predetermined objective, which is the transformation of raw material into

marketable products". Also, they refer to Neto (1996) by "a system should be considered complex when it is made of groups of elements with different functions and behaviors. They are in constant evolution and are influenced by events that cannot be foreseen with certainty. The information about the state of these elements cannot be completely known, and the elements are related by a great variety of interrelationships."
Next, they point out and discuss peculiar characteristics of complex systems:

- auto-organisations and emergence
- connectivity and environment
- non-linearity and feedback
- far-from-equilibrium
- structure and composition

In this discussion, they also mention Eijnatten (2003) who defined organisations as chaordic systems: systems composed of elements connected in a complex and dynamic form, forming a whole whose behaviour is simultaneously unpredictable (chaotic) and standardised (having an order).
Later in their paper they refer to Donnadieu et al. (2003), Neto (1996) and Capra (1996) for considering three dimensions when studying complex systems, productive systems or live systems:

- structural
- functional
- evolutionary

Fontana and Neto also describe organisations as "complex systems, formed by autonomous entities, interconnected in different ways and at different intensities. They are self-organizing and self-generating entities in higher levels. Their behaviour emerges as a result of the non-linearity of its feedback structures and its structures co-evolve with the environment, with the potential to generate a new order after periods of instability."
Fontana and Neto focused on the complexity changes of the organisations themselves caused by an ERP implementation process. However, they did not concentrate on the complexity of the ERP implementation process itself, or how to approach this complexity by research, as is our main research goal. Also, we found no additional research by these authors on this subject.

2. Ghosh, S., & Skibniewski, M. J. (2010). Enterprise Resource Planning Systems Implementation as a Complex Project: A Conceptual Framework. Journal of Business Economics and Management, 533-549.

Ghosh and Skibniewski discuss critical success factors and risk factors related to the complexity of ERP projects. They consider ERP implementation projects as complex adaptive systems. Unfortunately, they state that they will not give a definite definition of complex ERP projects and seek out explanation of complexity more in discussing dimensions or properties of the complexity of ERP projects, than in a definition itself. They discuss ERP projects considering four types of complexities: structural, technical, directional and temporal complexities provided by Remington et al. (2007). In their paper, they mention that ERP implementations are complex because they are systems. They quote Willems on what a complex system is: "one made up of a large number of parts that interact in a non-simple way. In such systems the whole is more than the sum of the parts, not in an ultimate, metaphysical sense but in the important pragmatic sense that, given the properties of the parts and the laws of interaction, it is not a trivial matter to infer the properties of the whole" (T. M. Williams, 1999). They also mention Baccarini's (1996) definition of project complexity as "consisting of many varied interrelated parts".

Although Ghosh and Skibniewski did not provide a definition of ERP implementation complexity themselves, they referred to the definitions of Willams (1999) and Baccarini (1996) for general definitions of complexity for projects.

Gosh and Sibniewski did discuss the complexity of the ERP implementation itself, but in a way that they tried to map CSFs and risk factors on ERP project complexity. However they did not discuss on a higher level how research should perform research, i.e. approach the complexity of ERP implementation.

3. Bradford, M., & Florin, J. (2003). Examining the role of innovation diffusion factors on the implementation success of enterprise resource planning systems. International journal of accounting information systems, 4(3), 205-225.

Bradford and Florin only refer to the definition of Rogers (2010) "Complexity is the degree to which a certain innovation is difficult to understand and use." This definition fits the term "complicated", which in common language is a topic which is difficult to analyse, understand, or explain.

Bradford and Florin did not discuss complexity or a complexity approach as is intended in our research.

4. Bollou, F., Balogun, E., & Usang, I. (2012). Eradicating complexity in software interface for increased productivity

Bollou et al. discuss the complexity of the user interface design of ERP systems. They explicitly discuss the construct of complexity by other researchers i.e. Manduca and Mogk (2006) and by Magee and de Weck (2004). From these definitions, they defined

the complexity of a system as an interaction between components or parts of a system that is difficult to comprehend which results in inefficiency for most users.
Bollou et al. did not discuss the complexity of ERP implementation projects itself or a complexity approach for ERP research.

5. Spiteri, K. J., Luca, C. L., Reynolds, T., & Wilson, G. Defining a baseline complexity model for ERP systems over SaaS. Journal of Internet Technology and Secured Transactions

With their research, they aim at defining software complexity itself within a business context for ERP systems over SaaS. They use ERP systems as an example of what could be considered a complex business system. By defining a measure of complexity, they aim at developing a deployment framework as a benchmark for the feasibility of complex systems on the Cloud. In this research applied elements of Complex Systems Theory, Network Complexity Theory and Programmatic Complexity Models to design a model for defining software complexity for ERP systems over Saas.
Unfortunately, they address in their research only the complexity of software itself and not the implementation process of ERP systems.
Spiteri et al. did also not discuss complexity of ERP implementation projects itself or a complexity approach for ERP research

3.2.4 Literature search conclusions

In our structured literature search, we strived at gaining insight into whether researchers explicitly use the construct of complexity as a consideration in the design of research into ERP implementation.
Out of our retrieved papers, we conclude that almost all papers use the words complex or complexity mainly as a quality related to ERP systems, ERP projects, ERP implementation and ERP environment. From the reviewed 117 papers, 113 did not define or explain what complex or complexity related to ERP implementation should stand for, or discussed ERP implementation within the context of complexity thinking. It seems that these authors use the words complexity as defined in http://www.merriam-webster.com (retrieved February 2015): “the quality or state of not being simple” or “the quality or state of being complex”. It seems a synonym for difficult to understand, which maybe better would fit the term complicated.
Only five of the 117 reviewed papers discussed the topic complexity further.
From these five papers, we had to conclude that three of them (Bollou, Balogun, & Usang, 2012; Bradford & Florin, 2003; Spiteri, Luca, Reynolds, & Wilson, 2012) did not discuss the complexity of ERP implementation projects itself. Only two papers (Fontana & Neto, 2009; Ghosh & Skibniewski, 2010) discussed complexity related to ERP implementation projects. Ghosh and Skibniewski concentrated on indicating what complexity of an ERP project is, but not how to approach this complexity in ERP implementation research.

Fontana and Neto (2009) did discuss the change of complexity of organisations by ERP implementations. Although they discuss complexity theory more extensively, they focused on complexity itself and also not on how to approach this complexity in ERP implementation research. The papers we retrieved which discussed complexity explicitly, all consider using complexity theory for research into ERP implementation useful. Nevertheless, the majority of the retrieved ERP implementation research uses complexity in a rather descriptive way instead of an analytical way.

As discussed in Chapter 1 and shown in this chapter, the construct of complexity and ERP implementation research are strongly linked. If complexity is an important topic in this research area, it should be explicitly be discussed when performing ERP implementation research, and also the research approach into this complexity should explicitly be chosen based on complexity theory. In our structured literature review, we have not come across research which explicitly based their research approach for ERP implementation on a specific complexity research approach. We consider only describing or defining complexity an insufficient base for performing structured research into the complexity of ERP implementation.

Therefore in our research, we will focus on how research can approach the complexity of ERP implementation. By explicitly thinking about complexity research approaches for ERP implementation research, we expect the results useful for performing more structured research into ERP implementation complexity. Also, these approaches might lead to new viewpoints on research into ERP implementation, as it may enable us to position existing research based on complexity and discover inconsistencies and incompleteness of ERP implementation research. However, in our literature, we did not discover a differentiation which was used to reflect explicitly on the complexity research approach into ERP implementation. We consider it essential to find and propose a suitable differentiation for research approaches into complexity.

To find explicit research approaches into complexity for ERP implementation, we consider it vital as a firm basis, first to discuss and elaborate complexity as a construct and relate it to ERP implementation. After that, we need to identify which theoretical approaches for research into ERP implementation complexity seem relevant for our research. The next sections discuss the construct of complexity and relevant research approaches into the complexity of ERP implementation.

3.3 Complexity as construct

3.3.1 Introduction

As a result of our structured literature review, we concluded that although research often mentions complexity related to ERP implementation, yet we did not retrieve research which specifically approached an ERP implementation project itself as a phenomenon by a scientific view on complexity.
To perform research according to a scientific complexity view on ERP implementation, we first need to clarify the construct of complexity itself. Although several authors did discuss characteristics of complexity and referred to definitions of other authors, we consider it essential in our research to start by discussing complexity as a construct. Therefore in this section, we will first define the construct of complexity in general. Also, as we are explicitly interested in the complexity of ERP implementation, we will base the definition of complexity of ERP implementation for our research on that definition.
However, merely a definition of a construct is insufficient for performing structured research. As our main research goal is to contribute to the understanding of the complexity of ERP implementation, we also wondered what research approaches could explore the complexity of ERP implementation in a methodical manner. We need a model/differentiation to look at complexity from different perspectives. Therefore after defining ERP implementation complexity, we will identify a differentiation for comparing different research approaches for ERP implementation research.

3.3.2 What is complexity and complexity of ERP implementation?

In this section, we will further explore complexity as a construct. First, we will introduce complexity definitions by literature and definitions encountered by our structured research review on the use of complexity on ERP research. Next, we will discuss the expression complexity in common parlance and research. Finally, we will present the complexity definition for ERP implementation we considered most appropriate for our research purposes.

Although, as shown in a previous section, frequently the expressions ‘complex’ and ‘complexity’ appear in the scientific literature concerning ERP implementation, they are rarely explicitly defined. Even in research on large change projects, where the generic construct of complexity is commonly used, hardly any effort is spent on a clear and explicit definition of complexity (Hertogh & Westerveld, 2009). In the five papers which more comprehensively discussed the complexity of ERP implementation, we also did not retrieve a clear definition of complexity for ERP implementation. In these papers, complexity is more or less defined by mentioning the elements or characteristics of complexity. Nevertheless, we will discuss how several of these authors and by them cited authors describe complexity.

MacCarthy (2000) considers organisations complex adaptive systems, consisting of an integrated assembly of interacting elements, designed to carry out cooperatively a predetermined objective, which is the transformation of raw material into marketable products.
Neto (1996) considers a system complex when it is made of groups of elements with different functions and behaviours. They are in constant evolution and are influenced by events that cannot be foreseen with certainty. The information about the state of these elements cannot be completely known, and a great variety of inter-relationships relate the elements to each other.
Fontana and Neto (2009) discuss as characteristics of complex systems: auto-organizations and emergence, connectivity and environment, non-linearity and feed-back, far-from-equilibrium, structure and composition. Based on Donnadieu et al. (2003), Neto (1996) and Capra (1996), they consider three dimensions when studying complex systems, productive systems or live systems:

- structural
- functional
- evolutionary

They process these insights into their definition of organisations as "complex systems, formed by autonomous entities, interconnected in different ways and at different intensities. They are self-organizing and self-generating entities in higher levels. Their behaviour emerges as a result of the non-linearity of its feedback structures and its structures co-evolve with the environment, with the potential to generate a new order after periods of instability."
Ghosh and Skibniewski (2010) considered four types of complexities: structural, technical, directional and temporal complexities provided by Remington et al. (2007).
Williams (1999) defined a complex system as "one made up of a large number of parts that interact in a non-simple way. In such systems the whole is more than the sum of the parts, not in an ultimate, metaphysical sense but in the important pragmatic sense that, given the properties of the parts and the laws of interaction, it is not a trivial matter to infer the properties of the whole".
Baccarini (1996) defined project complexity as: "consisting of many varied interrelated parts".
Magee and de Weck (2004) defined a complex system as "a system with numerous components and interconnections, interactions or interdependencies that are difficult to describe, understand, predict, manage, design, and/or change."
Hertogh and Westerveld (2009) compared complexity descriptions from thirteen authors: Robert Axelrod. Michael D. Cohen; Peter Coveney, Roger Highfield, Robert L. Flood, Ewart R. Carson, Murray Gell-Mann, Joel Moses, Charles Perrow, Eberhardt Rechtin, Mark Maier; Peter Senge; Ralph Stacey, John Sterman; Joseph Sussman; Geert Teisman and Edward O. Wilson.

Hertogh and Westerveld turned to Senge for a distinction into two perspectives on complexity:

1. Detail complexity
2. Dynamic complexity

Also, they showed that other authors use similar distinctions (Table 3.3).

Table 3.3 Distinctions in complexity made in theory (Hertogh & Westerveld, 2009)

Author	Description of types of complexity	
Senge (1994)	Detail complexity	Dynamic complexity
Teisman (2005)	Complicated	Composed
Whitty & Maylor (2007)	Structural	Dynamic
Williams (2002)	Structural	Uncertainty
Many Authors {i.e: Kurtz, Snowden)	Complicated	Complex

In these two perspectives: **components with their interrelations, evolvement over time and limited understanding and predictability**, are considered three characteristics of complexity, as shown in table 3.4

Table 3.4 Perspectives and characteristics of complexity (Hertogh & Westerveld, 2009)

Perspective	Characteristics
1. Detail complexity	o Many components with a high degree of interrelatedness.
2. Dynamic complexity	o The potential to evolve over time: self-organization and co-evolution. o Limited understanding and predictability.

Aside from Hertogh and Westerveld, these three characteristics can also be identified in the complexity descriptions from the discussed authors above. Also, these other authors do not show additional characteristics.

Table 3.5 shows which characteristics of complexity are mentioned by which author.

Table 3.5 Characteristics of complexity by author

Author	Characteristics		
	components with their interrelations	evolvement over time	limited understanding and predictability
(Hertogh & Westerveld, 2009)	✓	✓	✓
(McCarthy et al., 2000)	✓		
(Donnadieu et al., 2003), (Neto, 1996), (Capra, 1996)	✓	✓	✓
(Fontana & Neto, 2009)	✓	✓	✓
(Ghosh & Skibniewski, 2010), (Remington & Pollack, 2007)	✓	✓	✓
(T. M. Williams, 1999)	✓		✓
(Baccarini, 1996)	✓		
(Magee & de Weck, 2004)	✓	✓	✓

However, in general, these authors discuss only the characteristics of a complex "something", in most cases that "something" is a system or project. A definition of complexity as construct itself is not provided, as also was noticed by Hertogh and Westerveld (2009). Complexity is often used as a common language expression as a synonym for "something that is not completely understood" or something with a similar meaning but without any implication. We will discuss complexity as a construct from the common language view as well as from a philosophers view to establish a definition suitable for our research into the complexity of ERP implementation.

First, we will discuss complexity as used in common parlance. As shown before http://www.merriam-webster.com (visited 27 July 2015) defines complexity as: "the quality or state of not being simple" or "the quality or state of being complex", which we rather consider a synonym for difficult to understand or to be equal to complicated. The Cambridge dictionary defines complexity as: "When something has many parts and may be difficult to understand or find an answer to". The Oxford dictionary definition of complexity is: "The state or quality of being intricate or complicated", which merely uses synonyms to define the construct. Complexity in Wikipedia (visited 27 July 2015) is described by "Complexity is generally used to characterize some-

thing with many parts where those parts interact with each other in multiple ways." As shown by these definitions and numerous other similar definitions, a phenomenon has many parts and/or is difficult to understand. Certainly, in the daily us of the word complexity, it is used by someone as soon as he or she cannot completely understand something he or she is interested in or has to deal with. In common parlance, complexity is a qualitative label which indicates that something is difficult to understand and as a result difficult to manage.

In science, however, complexity is considered a construct which can be used to value phenomena in which science is interested. Unfortunately, in science, complexity can be defined in several ways, i.e. as a quantitative or a qualitative construct (Standish, 2008). In the "quantitative realm" complexity is used as an absolute measurable quantity property of a phenomenon, for instance, the processing time for solving an algebraic equation (Edmonds, 1999; T. Lee, 2003). As a quantitative construct, complexity is used to compare and predict the behaviour of a phenomenon. However, in other research areas, complexity as a construct is considered a qualitative predicate of a phenomenon. The view as a qualitative construct refers to emergent phenomena and their behaviour, which cannot simply be predicted by understanding just the structure of the phenomenon (Standish, 2008). Several authors demonstrate a variety and development of complexity theories in research (Anderson, 2013; Alhadeff-Jones, 2008). Several "complexity" researchers (Edmonds, 1999; T. Lee, 2003; Standish, 2008) also demonstrate that different research areas use complexity as a construct with various meanings in alignment with differences in corresponding research goals. Given this vast variety of research goals, it is understandable that an accepted shared understanding, meaning and theory of complexity has not yet emerged. Therefore the required depth in complexity research to solve this problem, largely exceeds the scope of this dissertation. However, if we confine ourselves to the challenges of ERP implementation, we may be able to narrow down to a simpler, but a still useful construct of complexity. Therefore we first will discuss whether the construct for ERP implementation complexity in our research should be regarded a quantitative rather than qualitative property of ERP implementation as a phenomenon.

In Chapter 1, we already discussed that current wisdom for ERP implementation largely depends on the ability to establish basic predictions on future costs and efforts based on extrapolation models, rather global lists of critical success factors and assumed best practices. Still, a significant success rate in ERP implementations has not been achieved, as ERP implementations tend to take longer than expected, cost more than expected and even more important, do not always meet the expectations about benefits and acceptance by the users. Commonly, it is assumed that most of the activities, resources and (end)products are sufficiently known at the start and during the implementation. However, observations of actual ERP implementations, as phenomena, show lots of sometimes threatening but always surprising dynamics, i.e. not predicted in the traditional views commonly known in ERP implementation literature and practices. For example, in many situations managerial awareness of potentially

changing stakeholder expectations is lacking, potentially causing harmful discussions between management and the workforce on the goals and the net results of an ERP implementation. Simple, straightforward quantitative prediction techniques are not suitable for identifying such lack of awareness. A qualitative view, on the other hand, might have. Arguably, some quantitative measurement will always be necessary for the phenomena that are well understood and predictable. For instance, in case an organisation holds a large quantity and variety of new ERP users it makes sense that the implementation is likely to be more complex than an organisation with only a few new ERP users with no variation. On the other hand, it is important to notice that an ERP implementation is a combination of technical and organisational changes which takes place in many dimensions and interacts with the surroundings of the organisation. The issues at hand in ERP implementations entail a multitude of factors which also require an in-depth qualitative view. These characteristics of ERP implementation compel us to adopt an accepted definition for complexity which should correspond to the qualitative complexity perceptions of an ERP implementation.
Fundamental research into the construct of complexity itself is a philosophical area. Hence we turn to that field of research for a suitable definition for our research to retrieve a base definition which considers complexity as a qualitative property of a phenomenon.
Edmonds (1999) as philosopher made a profound study of the construct of complexity itself through his thesis on the measures of complexity. He proposed a universal applicable working definition derived from numerous accepted but limited applicable definitions of complexity:

Complexity
'*That property of a language expression which makes it difficult to formulate its overall behaviour, even when given almost complete information about its atomic components and their interrelations.*'
(Edmonds, 1999)

This definition takes into account the three characteristics of complexity which Hertogh and Westerveld (2009) identified and other authors discussed before as shown in table 3.4. The characteristic "components with their interrelations" can be identified as atomic components and their interrelations. "Evolvement over time" and "limited understanding and predictability" can be considered as difficult to formulate its overall behaviour. The definition of Edmonds (as shown in his thesis) is general enough to apply to various areas.
Some illustrative examples:

- This definition can be applied to mathematical formulas. A formula can be considered complex when the design of that formula for a specific phenomenon can be very difficult. The three-body problem might illustrate that. The three-body problem is the problem of taking an initial set of data that specifies the positions, masses and velocities of three bodies for some particular point in time (for instance earth, moon and sun) and then determining the motions of these three bodies, in accordance with the laws of classical mechanics (Newton's laws of motion and of universal gravitation). Although the laws of classical mechanics are known (the atomic components), yet still an analytical solution is not possible only in special cases an analytical approximation. The three-body problem is considered a complex problem in physics. Complexity is considered a characteristic of the three-body problem.
- Another example is weather forecasting. For the purpose of weather forecasting, around the world weather stations measure in real-time local weather conditions and store historical data (the atomic components). With data and formulas, the weather is reasonable predicted for a few days in time. However, longer term predictions are inaccurate. Weather as a phenomenon is very difficult to understand and to predict. Therefore complexity is a property of weather forecasting.
- Edmond's definition also can be applied to social phenomena. For instance, in election times, lots of data are collected about the preferences of voters. A vast amount of historical data about past elections is available. Also, the demographic map is accurately known. Still, the outcomes of elections are sometimes very surprising, for instance, Donald Trump's election as president of the USA. It can be considered a complex phenomenon.

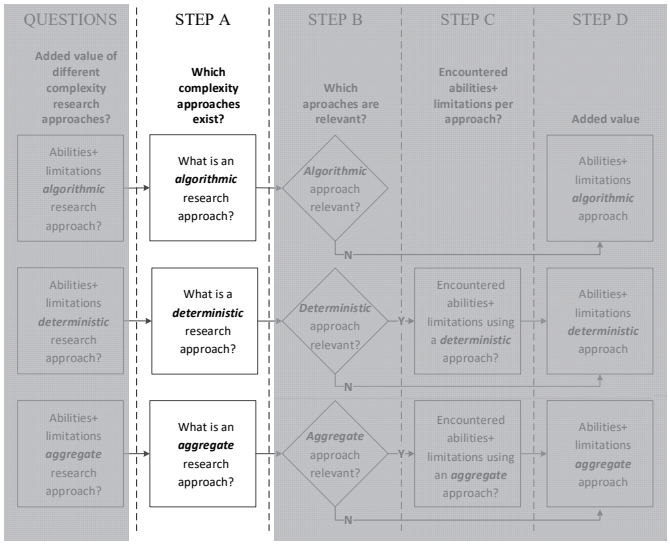
Our research aims at enhancing understanding of complexity research into ERP implementation. To do that, we need to define what we consider ERP implementation complexity.
In Chapter 1 we discussed our definition of the phenomenon ERP implementation for our research:
All activities undertaken, resources needed, (sub)products produced, stakeholders, and their interrelationships to introduce (parts of) an ERP information system in an organisation and the associated necessary organisational changes.

Based on the definition of complexity by Edmonds (1999) and our research definition of ERP implementation, we define the complexity of an ERP implementation for our research as follows:

Complexity of an ERP implementation
That property of an ERP implementation which makes it difficult to formulate its overall behaviour, even when given almost complete information about its activities, resources, (sub) products, stakeholders, their interrelations and the associated necessary organisational changes.

3.4 Complexity research approaches

3.4.1 Introduction



In the previous section, we aimed at achieving a suitable definition for the construct of the complexity of ERP implementation. However, merely a definition is insufficient for performing structured research into a construct. We intend to approach research into the complexity of ERP implementation itself in a systematic manner. However, complexity in science is discussed and researched in numerous ways in different disciplines, from fundamental philosophy to physical sciences. Also in ERP implementation research, which intends to contribute to a better understanding and managing of the complexity of ERP implementation, different research approaches exist. For instance, approaches by design of formulas which can predict the time, effort and cost of ERP implementation projects by analysing historical data (Arb, 1997; Francalanci et al., 2001; Stensrud, 2001). Approaches by collecting critical success factors by studying cases and asking expert's opinions (Al-Mashari, Al-Mudimigh, & Zairi, 2003). Also, some research approaches the complexity of ERP implementation by gaining more insight into the structure of the project, by researching into implementation phases, activities and players (Ruivo, Oliveira, & Neto, 2014; Toni M Somers & Nelson, 2004). These approaches implicitly presume a certain view on complexity and how research might contribute to handling this complexity. Although as far as we were able to identify, their research does not explicitly discuss the adopted views.
If we can distinguish different views on complexity, then we can, based on these views, assess the abilities and limitations of research approaches into the complexity of ERP implementation in a methodical manner. After all, our main research question

is: "What is the added value of explicit application of different complexity research approaches into ERP implementation?". To be able to distinguish different views, we considered that a differentiation for separation of different complexity research approaches into the construct of complexity could be suitable. A suitable differentiation can also be used in future research into complexity to be able to apply a view on complexity explicitly and thereby better be aware of the limitations of the research and results.
Of course, we prefer a well-known differentiation. Therefore we searched for differentiation of research approaches into complexity, which is clear, practical and well-known. We encountered Manson's differentiation, which appears to be a well-known model in the complexity research area as it is referred to about 500 times. Manson distinguishes only three general approaches which are easily understandable. We considered this number of approaches practical for our research purposes. Also, he clearly separates and discusses these distinct approaches and describes characteristics of each approach. Therefore satisfying our requirements clearness, practicality and well-known, we chose to adopt the complexity research approaches as proposed by Manson (2001) as the basis for our research.

Manson distinguishes the next approaches:

- algorithmic complexity approach
- deterministic complexity approach
- aggregate complexity approach

We are aware that other differentiations of approaches might exist in the extensive scientific field of complexity theory research. Moreover, there also exists a wide variety of complexity theory research in various areas and no generally accepted complexity theory exists (Alhadeff-Jones, 2008). We adopted Manson's three approaches because we consider them clear and practical for our research.
Although we chose Manson's differentiation for or research, nevertheless in Chapter 6 (Conclusions) we will discuss if this differentiation was meaningful for our research and future research.

To be able to answer our main research question: "What is the added value of explicit application of different complexity research approaches into ERP implementation?", we designed our research around these three complexity approaches. In other words, we chose these three complexity approaches as a foundation for our research. We explored these three complexity approaches theoretical and by empirical research to be able to answer our main research question.

In the next sections, first, we will discuss the three complexity approaches in general as proposed and described by Manson. After that we will discuss the relevance of

each of these complexity approaches for our ERP implementation complexity research, i.e. is every approach worthwhile investigating in our research.

3.4.2 Algorithmic complexity approach

The algorithmic complexity approach takes the view that a simplest computational algorithm can be designed, which can reproduce system behaviour. It assumes that a set of mathematical algorithms can model the system and that these algorithms can reproduce system behaviour, which of course can be used to understand and control its behaviour. This complexity approach is based on a strong objectivistic and positivistic worldview. Algorithmic complexity approaches presume that algorithms can express every behaviour of a phenomenon. In this approach, a phenomenon is regarded more complex when it is more difficult or even impossible to model all necessary algorithms, and/or it is difficult or even impossible to compute these algorithms. The phenomenon for which the complexity is modelled is considered a system which consists of known essential elements and their relationships by which the behaviour of the system can be modelled. If a phenomenon is correctly modelled, the outcome of the algorithms will be suitable to understand and control its behaviour. An example of an algorithmic complexity approach is Newton's laws of motion. By these laws, the relationship between a body, the forces acting on it and its motion in response to those forces, can be understood and predicted. The more bodies and forces exist in a situation; the more complex the phenomenon will be considered, and the effort for formulating the correct algorithm and calculating its behaviour will increase. The algorithmic complexity approach strives to understand a phenomenon and finding the algorithm to that extent which explains and predicts the behaviour of that phenomenon correctly.

3.4.3 Deterministic complexity approach

In contrast to an algorithmic complexity approach, a deterministic complexity approach uses a different view on complexity. It does not take the position that a phenomenon itself can be fully understood and modelled (entirely expressed in matching mathematical algorithms) since the phenomenon is perceived a chaotic system. Nevertheless, the system has an underlying order, but this order is too complicated to be comprehended and modelled. However, by the discovery of a few key variables related through a set of confirmed equations, it is assumed that the behaviour of the system can satisfactorily be described and the complexity of its behaviour be understood. Discovery of these key variables and design of these equations is a difficult task. Confirmation of this deterministic complexity of a system can only be made by a large amount of time series data (Manson, 2001).
Research into medication is a good example of this deterministic complexity theory type. The positive and negative effects of a drug on the human body is never completely understood, as every individual has a unique body and mind and also exists

in a unique environment. Besides that, also the operation of the human body is certainly not yet fully known. Nevertheless, medical science designs key variables and equations to develop medication and determination for proper use of medication. The limited understanding of the operation of the human body still leads to testing the impact of certain drugs on a representative population for the development of useful medication. Large amounts of time series data are needed to verify the usefulness and safety of the drug. However, problems with negative side effects of medications also show that the behaviour of the body as a system was (and undoubtedly still is) not satisfactorily understood. For instance thalidomide, a sedative and hypnotic drug was, despite testing of the drug, withdrawn from sale in the early 1960s after it was found to cause severe congenital disabilities when taken during pregnancy. More than 10,000 children in 46 countries were born with deformities such as phocomelia as a consequence of thalidomide use.

Manson discussed four characteristics of deterministic complexity:

1. Deterministic mathematics and attractors
2. Feedback
3. Sensitivity to initial conditions and bifurcation
4. Deterministic chaos, strange attractors and fractals

We will briefly discuss Manson's characteristics for his deterministic complexity.

Deterministic mathematics and attractors
Simple mathematical terms should be able to describe and potentially understand chaotic or catastrophic systems. A few key variables related through a set of known equations can describe the behaviour of a complex system in a useful manner. Also, Manson defines attractors as: "values towards which system variables tend to settle over time". For example, the growth of the population of a certain species of animals can be by predicted by a formula which contains variables about a current number of individuals, a growth rate parameter, etc. This formula is a simple equation with a few key variables. This formula cannot predict the exact growth of the population. However, the outcome of the calculations is sufficiently useful for the planning and managing of the habitat.
However, if the population of a certain species of animals gets too small, this species eventually will become extinct. This extinction can be considered an attractor.

Feedback
When using the deterministic complexity approach, these simple mathematic equations of deterministic complexity allow for dynamic behaviour by incorporating feedback. For instance, in the animal species example, the calculation of the future population is dependent on the current population.

Sensitivity to initial conditions and bifurcation
Manson discusses that under certain circumstances, a system is sensitive to initial conditions and small changes in the initial system may lead to large, non-linear effects. For instance, the butterfly effect exemplifies this sensitivity to initial conditions. The flapping of a butterfly in one part of the world may majorly influence weather in another part of the world.
The characteristic of a system for jumping suddenly from one attractor to another is called bifurcation. This bifurcation characteristic applies especially to catastrophic systems. However, in science, catastrophic attractors are mainly suited for the modelling of natural phenomena and are less applicable to social systems.

Deterministic chaos, strange attractors and fractals
Manson explains with deterministic chaos that if variables in a deterministic equation have certain values, then the predicted behaviour of the system will be seemingly randomly behaviour (chaotic behaviour). Also, there may be strange attractors, which means there will be values of variables by which the system will almost reach a certain behaviour, but never quite. Next Manson discusses fractals. Systems can show behaviour as fractal patterns. The advantage of understanding fractal behaviour patterns from the system at one scale may lead to understanding the systems behaviour at other scales. For instance, the structure of trees (branches which fork again and again) shows the same behaviour on a detail level as on a more global level.

3.4.4 Aggregate complexity approach

Equal to algorithmic and deterministic complexity approaches, the aggregate complexity approach considers a phenomenon a system. However, the aggregate complexity approach uses a much broader view on the system and its complexity. Algorithmic and deterministic complexity approaches rely on mathematical equations and a number of assumptions about how systems work. Aggregate complexity approaches instead, attempt to access the holism and synergy resulting from the interaction of system components (Manson, 2001). Manson discussed aggregate complexity as a set of interrelated concepts that define a complex system:

- Relationships
- Internal structure
- Environment
- Learning and memory
- Emergence
- Change and evolution

We will discuss these concepts:

Relationships

A system consists of components and their relationships as symbolised in figure 3.1. The circles in this figure represent components, the connecting lines their relationships and the dotted ellipses sub-systems.

According to Manson "A complex system is defined more by relationships than by its constituent parts" (Manson, 2001). For instance, the intensive exchange of goods and capital (relationships between companies) will influence the economy more than the sheer existence of companies, goods and capital (components). Also, through these relationships sub-systems will form with their specific functions and goals. These sub-systems do not even have to be aware of the goals and functions of the complete system.

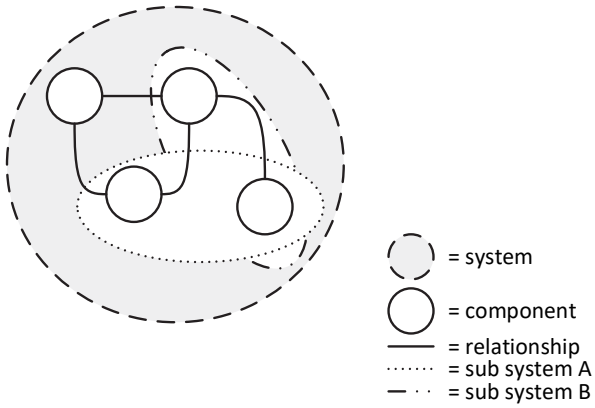


Figure 3.1 Components, relationships and subsystems

Internal structure

Not only the existence of components and relationships defines the behaviour of a system but also the different types of components and different types of relationships between components as shown in figure 3.2. The total of these components and relationships and their different types forms the internal structure. Different shapes of symbols represent the different types of components, and the different types of lines between these symbols represent the different types of relationships.

For instance, different types of relationships on an individual will influence individual's behaviour differently. A personal relationship, like a marriage, will influence an individual in a different way than a professional relationship like an employment relationship. Also, a component of the type female will, in general, behave differently than a male type component.

Components can form part of multiple sub-systems through their different types of relationships with other components.

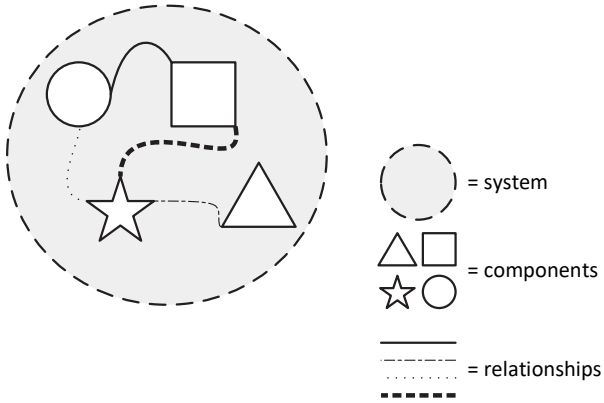


Figure 3.2 Different types of components and relationships

Environment

A system is always embedded in an environment and has relationships with that environment. Thus, an organisation will always exchange information, goods and energy with its environment and this exchange will influence an organisation as figure 3.3 symbolises. The dotted circle represents the system and the different lines the relationships with its environment. For instance, an organisation will have relationships with the government.

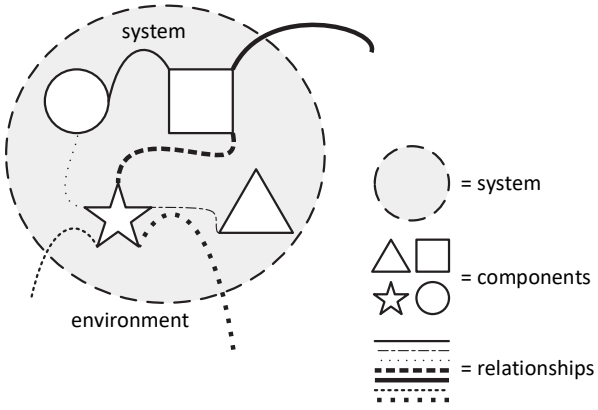


Figure 3.3 Relationships with environment

Learning, memory, change and evolution

Influenced by its environment a system shapes, reacts and anticipates, which can be regarded as "learning" from the past. To be able to "learn" a system needs to remember what happened in the past, its reaction and outcome. A system also has the ability

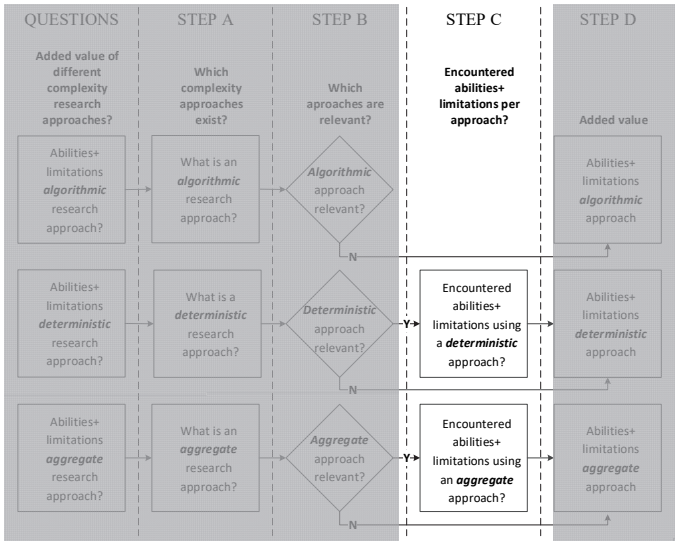
to react to novel situations. These abilities can already be present by the possibilities of the components and sub-systems or have to be created by self-organization. A system can form new relationships and change components, as a reaction to its past or present. Also, these new relationships and changed components can form new subsystems. For example, an organisation may react to a competitor with a faster delivery time by optimising the relevant business processes (relationships) and training the personnel (components) who performs these business processes, thus changing its components. A system can change its internal structure and behaviour, possibly forced by its environment, to interact better with its environment. This change can occur gradually or abruptly via a highly unorganised state. For instance, new government laws may force organisations to abruptly change their ways of doing business or even lead to bankruptcy if they cannot change in time.

Emergence

"The capacities of a complex system are greater than the sum of its constituent parts" (Manson, 2001). A system may show behaviour which is not expected by the existing knowledge of its internal structure. Suppose the following rather ideal situation; all functionality of the software of the social media is known. Also, the participants in these social media and their relations are stored in the databases of the software. Furthermore, understanding of human interactions in social media exists. Despite this, it cannot be predicted or controlled how the participants will interact with each other and what new social phenomena will emerge from this interaction through these social media. For example, the phenomenon "flashmob" emerged from the possibilities of the social media, but could not be foreseen. Flashmob is a group of people who assemble suddenly in a place, perform an unusual and seemingly pointless act for a brief time, then disperse, often for the purposes of entertainment, satire, and artistic expression (Wikipedia status 26 March 2013).

3.5 Relevant complexity research approaches for ERP implementation

3.5.1 Introduction



In the previous section, we introduced Manson's differentiation for complexity approaches. In this section, we will discuss which of Manson's complexity approaches we considered relevant for further exploration for our research into ERP implementation complexity. First, we will discuss what type of Manson's complexity research approaches we encountered in the reviewed 117 papers from our structured literature search discussed in section 3.2, to get an indication which approaches might prevail in ERP implementation research. Next, we will discuss every complexity approach by a comparison of the characteristics of that approach as discussed in the previous section with characteristics of ERP implementation. By this comparison, we will argue per complexity approach whether that approach seemed relevant for further empirical exploration in our research.

3.5.2 Prevailing complexity research approaches in current research

After selecting Manson's three complexity research approaches as a basis for our complexity research differentiation, we were interested in what complexity research approaches current research uses. However, studying every present ERP implementation research paper to determine what complexity research approaches prevail, would be impossible due to the vast amount of existing ERP literature. Also, we concluded from our structured literature review that research we retrieved about ERP imple-

mentation and complexity, did not explicitly discuss their view on complexity and therefore their complexity research approach on ERP implementation. Because we carefully selected these papers, which specifically discussed ERP implementation and also discussed complexity as a topic, we expect that the complexity research approach in other ERP implementation literature is even less likely to be explicitly discussed. Therefore our purpose was to gain a proper indication of the prevailing complexity research approaches within the ERP implementation research field, even if not explicitly discussed in that research. We assumed that by analysing our carefully selected 117 papers on ERP implementation and complexity, we might gain that proper indication. These 117 papers could provide us with an indication of the distribution of the implicitly used complexity research approaches. I.e., which of Manson's complexity research approaches is implicitly most commonly employed by ERP implementation research?

Therefore we studied every paper of our reviewed 117 papers with the characteristics of Manson's approaches in mind, as in our structured literature review we selected every paper because it discussed performed research into ERP implementation. Based on Manson's definitions and characteristics of the three research complexity approaches as discussed in section 3.4, we classified each paper into one of the three approaches. On researcher performed this rating. This researcher studied the content of each paper in detail and decided by Manson's characteristics which complexity research approach that paper implicitly used, by keeping in mind the description of Manson for that approach and its qualitative characteristics as discussed in the previous section.

Although a cross-checked rating by other researchers would reduce bias, we felt for our purpose of gaining an indication, rating by only one researcher would suffice. Our goal was to gain an idea of the distribution of the implicitly used complexity research approaches within current ERP implementation research and not an accurate distribution of these 117 papers.

All 117 papers, were rated as performed by a deterministic complexity research approach.

Therefore the outcome of this rating might suggest that in current research on ERP implementation, the deterministic complexity research approach implicitly prevails.

3.5.3 Relevance of an algorithmic complexity research approach

An algorithmic complexity approach assumes that there is a simplest computational algorithm that can reproduce system behaviour.

To design this algorithm, one has to be aware of all conditions and surroundings of the phenomenon, i.e. a complete understanding of the phenomenon. Also, Manson (2001) argues that when applying algorithmic complexity research approaches to social or environmental phenomena it may incorrectly equate data with knowledge, as there are important aspects that cannot be expressed in algorithmic expressions. For

instance, a mathematical algorithm cannot model straightforwardly human experience, personality, groupthink et cetera, which nevertheless, can influence the behaviour of the phenomenon.
By researching ERP implementation complexity by an algorithmic approach, two problems may arise. Firstly an ERP implementation consists of too many elements, types of elements and relationships between these elements, which makes it practically impossible or useless to compose a comprehensive set of algorithms that will represent the implementation. Secondly, an ERP implementation is also a social phenomenon in an organisation (Kwahk, 2006). As can be retrieved from critical success factors for ERP implementation (Dezdar & Sulaiman, 2009), objects like people, technical infrastructure, software, business processes, data, legacy systems etcetera all influence ERP implementations in a positive or negative way. These objects all relate and influence each other. It would be unrealistic to assume that an algorithm could be designed, which could predict the influence of the behaviour of all these objects on an ERP implementation.
Therefore we consider an algorithmic complexity approach limited when used as a principle for understanding the actual complexity of ERP implementation.
Also as shown in the previous section we did not encounter in our 117 papers on ERP implementation and complexity any research which we considered based on an algorithmic complexity research approach.

We will not imply that an algorithmic approach could not have any value for ERP implementation research. However, for the reasons we discussed, we did not further explore the algorithmic complexity approach in our research. We expected that further exploration by empirical research would contribute too little to answering our main research question.

3.5.4 Relevance of a deterministic complexity research approach

Manson considers a deterministic complexity research approach still difficult to apply to social phenomena and this approach works best in natural science and physical geography (Manson, 2001). An explanation could be that social systems and behaviour of humans are continually subject to major unexpected and unnoticed changes. The phenomenon as a social system, therefore, is continually changing. Whereby the variables and equations which correctly represented the complexity of the system at a specific point in time have a high probability of being obsolete at a later point in time. Also, the fundamental question arises whether social systems can in principle be compared to each other? For instance, variables and equations which can reasonably predict economic growth in one country probably cannot be used in another country if their principles extremely differ, for instance socialism against capitalism or existence of different religions.

Nevertheless, in ERP implementation research these deterministic complexity research approaches provide useful predictions for ERP implementations. Although the nature of the designed prediction of management method itself carries a potential risk of being obsolete as soon as ERP implementation as a phenomenon changes its underlying logic or structure. Also, a limitation is that the quality of these deterministic based ERP prediction methods can only be secured by a continuous supply of large amounts of accurate and reliable historical data. Methods derived from this deterministic complexity research approach are always based on observations from the past. Decisions in an actual ERP implementation derived from these deterministic based ERP prediction methods will be based upon past events from other implementations. An analogy is driving a car by only looking in the driving mirror. As long as the road ahead will not deviate much from the road behind, driving will be no problem and only require some mild corrections. However, as soon as the road deviates from the road behind, for instance as soon as a crossing comes up, a driver cannot adequately respond and only after an accident has happened the driver will know what the correct respond should have been. ERP implementation models based on the deterministic approach consequently always will bear a risk. That is to say that use of their proven deterministic variables and equations in an actual ERP implementation might lead to problems in case ERP implementation as a phenomenon has seriously changed since the historical data, which were used to construct the deterministic variables and equations, were retrieved. New development and innovation cannot be accounted for in most deterministic type models. The users of the model also not directly detect when a system has changed. In the algorithmic complexity type, it will be immediately notified if the system does not behave in correspondence with the predicted behaviour. However, in the deterministic complexity type, a deviation from the predicted behaviour could not be notified immediately as there always will be an expected deviation from the predicted behaviour of the phenomenon.

In the deterministic research complexity approach in most cases, it is difficult to retrieve the required historical data for ERP implementations. Furthermore, the validity of these data can be uncertain, as politics in large projects without a doubt has influence on the completeness and correctness of the recorded data.

Another consideration when using a deterministic complexity research approach for analysing ERP implementation complexity is the fact that the technical basis of ERP information systems themselves has changed profoundly in the last 15 years, and diversified into a number of applications (Koch, 2011). Koch argues that "ERP-research is disregarding the profound software changes and their impact on the user context, when implementing and operating ERP." For instance SAP ERP software transformed during the period 1990 until the present from mainframe software to client-based software to cloud-based software. As a consequence, the implementation in organisations will have changed during that era, and models which are based on ERP implementation data from the mainframe era will probably not correctly represent an ERP implementation in the cloud-based era.

Despite these considerations, nevertheless as shown by current research and use of these results in practice, research based on this deterministic complexity approach provides useful results for understanding and controlling issues in ERP implementations.
In our discussion of the implicitly used complexity research approach for the 117 papers from our structured literature review we concluded that the outcome of our rating might suggest that in current research on ERP implementation, the deterministic complexity research approach implicitly prevails.

Derived from the above we consider in table 3.6 the abilities and limitations for application of a deterministic research approach for research into ERP implementation.

Table 3.6 Abilities and limitations of a deterministic complexity research approach for research into ERP implementation

Abilities	Limitations
o This research approach prevails in current ERP research and has retrieved useful results. o Methods and research areas for this type of complexity research approach into ERP implementation are well-known. o A vast amount of this type of research is available as a base for future research.	o It is difficult to model the behaviour of social phenomena like ERP implementations. o It is difficult to take into account unexpected changes in the ERP implementation field, by which the research results may become obsolete. o It can be difficult to retrieve historical data about ERP implementations required for keeping your insight up-to-date.

To further explore the relevance of the deterministic complexity research approach for ERP implementation, we will discuss for every characteristic Manson discussed for a deterministic complexity approach, whether ERP research can demonstrate these characteristics. As several characteristics are more appropriate for physical chaotic systems, not every characteristic can be linked to by ERP research.

Manson distinguished the following deterministic complexity research approach characteristics:

- Deterministic mathematics and attractors
- Feedback
- Sensitivity to initial conditions and bifurcation
- Deterministic chaos, strange attractors and fractals

Deterministic mathematics and attractors

As discussed by Manson (2001), simple mathematical terms should be able to describe and potentially understand, chaotic or catastrophic systems. In ERP research several authors proposed the "size of an ERP project" as a candidate for measuring the complexity of an ERP implementation project. With this "size" a formula could be found to predict the time, effort and cost of ERP implementation projects (Arb, 1997; Francalanci et al., 2001; Stensrud, 2001). They proposed several key variables for this "size" for example: 'number of users', 'organisational size', 'configuration size' and 'technical size'. Another set of models to predict ERP software engineering effort as well as total integration effort and the duration of the ERP implementations is based on measuring regarding the number of reports, interface, conversion, and extension (RICE) objects (Rosa, Packard, Krupanand, Bilbro, & Hodal, 2013). Other authors tried to combine methods and variables from software development, for example, COCOMO, into key variables and equations which can be used to define and predict the ERP implementation complexity (Daneva, 2010; Hansen, 2006; Magnusson et al., 2004).

Davidson (2010) discusses that in social systems attractors in nonmathematical terms can be viewed as a stable mode of behaviour. In the absence of disturbance, the system will tend to this stable mode of behaviour. However, if this social system is exposed to too many perturbations, the system might settle for a new attractor or even turn into chaos. In ERP implementations the project will continue to progress in a planned way as long as there are not too many perturbations. Risk management and buffers in time and budget will cope with perturbations of the project. However if a major perturbation will occur which cannot be smoothed by these safety nets, the project will have to be planned and budgeted again or even can fail.

We conclude that the characteristic deterministic mathematics and attractors of Manson's deterministic complexity approach can be linked to ERP research.

Feedback

When using the deterministic complexity approach, these simple mathematic equations of deterministic complexity allow for dynamic behaviour by incorporating feedback. In ERP implementations, as in every large projects, project management will adjust the initial planning based on feedback about the actual progress of the project. Information about the progress will be used as feedback for the planning of the re-

maining activities and cost of the project. Therefore this characteristic of a deterministic system can be linked to ERP implementation.

Sensitivity to initial conditions and bifurcation
A system is sensitive to initial conditions and small changes in the initial system may lead to large, non-linear effects.
Research into critical success factors for ERP implementation can be considered research into the sensitivity to initial conditions. In the ERP research field, much research has been performed and still is, into discovery and verification of critical success factors (Huang, 2010).
Hsu, Wang et al. (2009) developed a model to predict success/failure values to enable organisations to decide whether to initiate ERP, inhibit adoption or take remedial actions to increase the possibility of successful ERP implementations.
This research links the characteristic 'sensitivity to initial conditions'. In our overview of ERP implementation research we could not detect research into bifurcation (suddenly jumping from one attractor to another). However, as Manson discussed, in science these catastrophic attractors are mainly suited for the modelling of natural phenomena and are less applicable to social systems.

Deterministic chaos, strange attractors and fractals
We could not reveal the deterministic chaos characteristic (seemingly randomly behaviour) in ERP implementation research. Although maybe the complete failure of an ERP implementation and for instance bankruptcy of the organisation by this failure could be considered deterministic chaos.
Strange attractors, which means there will be values of variables by which the system will almost reach a certain behaviour, but never quite. In ERP implementations this could be interpreted as: an ERP system never will completely satisfy the needs of the organisation.
Fractal patterns: the advantage of understanding fractal behaviour patterns from the system at one scale may lead to understanding the systems behaviour at other scales. In general, large ERP implementations consist of subprojects. Management and controlling of these subprojects are in most cases similar to the overall project. Also if the subprojects are on schedule and budget, the overall project is also considered on schedule and budget.
Similar to the bifurcation, we consider these deterministic characteristics a bit farfetched for ERP implementation and more applicable to natural phenomena than to ERP implementation.

Considering linking of Manson's characteristics of deterministic complexity to ERP implementation and the expected prevalence in current ERP implementation research for a deterministic complexity research approach, we conclude that a deterministic complexity research approach is useful for ERP implementation research.

Therefore it seems worthwhile to further examine the abilities and limitations of applying a deterministic complexity research approach to ERP implementation research. In our empirical research, which we will discuss in Chapter 4, we will discuss own research by a deterministic complexity research approach to examine the abilities and limitations of this complexity research approach.

3.5.5 Relevance of an aggregate complexity research approach

As explained in section 3.3, an aggregate complexity research approach relies on a broad view on a system and does not rely on mathematical equations and assumptions of how a system works. If we relate that to ERP implementation, research with the intention to detect critical success factors, design prediction methods and also attempts to map the structure of an ERP implementation are not research views which fit an aggregate research approach.
The body of thought about an aggregate approach considers the behaviour of a system, in this case an ERP implementation (project), a phenomenon which will always have unexpected twists and turns. These twists and turns originate because of influences by the environment, the fact that the system learns, has a memory and maybe most important: a system changes during its lifecycle. These are clearly aspects which do fit social phenomena. ERP implementations change organisations as can be shown by case studies (Alhirz & Sajeev, 2015; Jääskelainen & Pau, 2009; Yeh & OuYang, 2010) but also numerous research exists which investigates change management aspects of ERP implementation (Grabski et al., 2011). An ERP implementation is also a social phenomenon and therefore it can be expected to show unexpected behaviour, which in practice means, the project can proceed differently than planned and even the outcome can be differently than expected (Burg van der et al., 2013). If an ERP implementation can show unexpected behaviour, then collecting vast amounts of historical data for retrieval of general guidelines, might introduce unreliable guidelines.
In our discussion of the implicitly used complexity research approach for the 117 papers from our structured literature review, we did not find research which used an aggregate complexity research approach. Apparently, current research does not commonly use this research approach. Therefore it is remarkable that ERP implementation possesses characteristics that at first sight fit an aggregate complexity approach, but research seems to apply mostly a deterministic complexity approach.

Derived from the above we consider in table 3.7 the abilities and limitations for application of an aggregate research approach for research into ERP implementation:

Table 3.7 Abilities and limitations of an aggregate complexity research approach for research into ERP implementation

Abilities	Limitations
o It is suited for research into the behaviour of social phenomena like ERP implementations. o It does take into account the unexpected behaviour of ERP implementations.	o This research approach is not yet explicitly used in current ERP research, and its usefulness is not yet demonstrated. o Methods and research areas for this type of complexity research approach into ERP implementation have to be developed. o Research is less focused on understanding a system entirely than on dealing with uncertainties.

As shown before, Manson discussed aggregate complexity as a set of interrelated concepts that define a complex system:

- Relationships
- Internal structure
- Environment
- Learning and memory
- Emergence
- Change and evolution

As an ERP implementation can be regarded a system, we will discuss these concepts and relate them to ERP implementation.

Relationships
When we apply this concept to ERP implementation, it is evident that a large number of components and especially relationships have to be considered in an ERP implementation. For example, the ERP system itself consists of various modules, connected to each other. Various users within the company will use these modules. The ERP system will be implemented on a specific IT infrastructure. IT professionals as stakeholders must construct this infrastructure. Other stakeholders (users, managers, clients et cetera) will use the ERP system to perform business processes. Top management wants business processes optimal supported by the ERP system to be able to meet their defined business goals. These are all examples of components and a large number of relationships between components that are present in ERP implementation.

Internal structure
ERP implementations usually deal with a variety of technical, organisational and interpersonal matters. The components of an ERP implementation will vary considerably. For example, ERP implementations have to deal with different types of stakeholders (Toni M Somers & Nelson, 2004). Some future users will be using the system for their primary work, for instance, checking invoices and other types of users which will be using the ERP system for providing necessary management data. Also, subsystems clearly exist in ERP implementation routes. For instance, an ERP implementation project is usually seen as consisting of subprojects with their specific goals and products (Daneva, 2010). Participants to the ERP implementation take part in one or more subprojects and in most cases also perform their regular job tasks, which can lead to opposing interests. The structure of an ERP implementation consists of a vast number of different types of components and relationships.

Environment
An ERP implementation always takes place in one (or more) organisations. The organisation influences the ERP implementation itself, but the implementation also influences the organisation before, during and after the implementation. In ERP implementations the implementing organisation also has to deal with the demands laid upon by their external relationships like suppliers and buyers. The functionality of the ERP system must be tailored to communication demands from suppliers and buyers. Also, government rules and culture must be respected (Alhirz & Sajeev, 2015; Krumbholz & Maiden, 2001). Besides that, the actions and abilities of the competitors have to be taken into account when implementing an ERP system. Even implementation of an ERP system by a competitor may initiate the implementation of an ERP system. Thus the environment of an ERP implementation plays a major role in decision making about and in the actual implementation of ERP systems.

Learning, memory, change and evolution
In ERP implementation clearly learning behaviour exists (Kraemmerand, Moller, & Boer, 2003). Organisations learn from previous implementations (Scott, 2000). They also learn during an actual implementation when parts of the system are established, the corresponding business processes are designed, and parts of the future user population are trained (Robey, Ross, & Boudreau, 2002). Implementation of the next parts will be based on the experience gathered during the previous parts, which is preserved in the memory of its participants, stakeholders and results, for instance, documents about the implementation. Also, users learn during working with prior implemented parts. Every ERP implementation is unique, therefore learning during implementation is always part of an implementation process.
In ERP implementation change and evolution also can be identified. In general, ERP implementations run over an extended period. Market demands, government demands but also internal changes like reorganisations, can change the initial demands

on the ERP system. If an ERP implementation did not adapt to these changing demands, the result would be a system that does not support the needs of the organisation. Another example of evolution during ERP implementation could be the fact that the ERP implementation started in some cases as an IT project, whereas the organisation realises during the implementation process itself that the ERP implementation is a positive organisational change and by implementing the ERP system this organisational change is achieved. Also during the ERP implementation process organisations can become aware that the implemented system will lead to new business opportunities.

Emergence
In ERP implementation the possibilities of the still to implement ERP system can lead to changed ideas about the operation of the organisation. Users discover new possibilities for the use of the system which were not possible with the legacy systems and demand to include these into the implementation path.

Every concept from Manson's interrelated concepts which characterise aggregate complexity seems to be able to link to ERP implementation easily. We considered the characteristics of the previously discussed deterministic research approach and concluded that a deterministic approach is mainly used in our retrieved ERP research into complexity. However, in our search for the construct of complexity in ERP implementation literature, we found no studies which seemed to have applied an aggregate complexity research approach.
Fontana and Neto (2009) discussed characteristics of complex systems: auto-organisations and emergence, connectivity and environment, non-linearity and feedback, far-from-equilibrium, structure and composition. Ghosh and Skibniewski (2010) used the definition of Williams: "one made up of a large number of parts that interact in a non-simple way. In such systems the whole is more than the sum of the parts, not in an ultimate, metaphysical sense but in the important pragmatic sense that, given the properties of the parts and the laws of interaction, it is not a trivial matter to infer the properties of the whole" (T. M. Williams, 1999). Nevertheless, Fontana and Neto and also Gosh and Skibniewski did not perform their research with an aggregate research complexity approach in mind.
Considering that the characteristics of an aggregate complexity approach fit ERP implementation complexity and we did not discover studies performed with an aggregate complexity view in mind, besides the deterministic approach, it seemed relevant to explore further whether also a research aggregate complexity approach is a feasible approach for ERP implementation research.

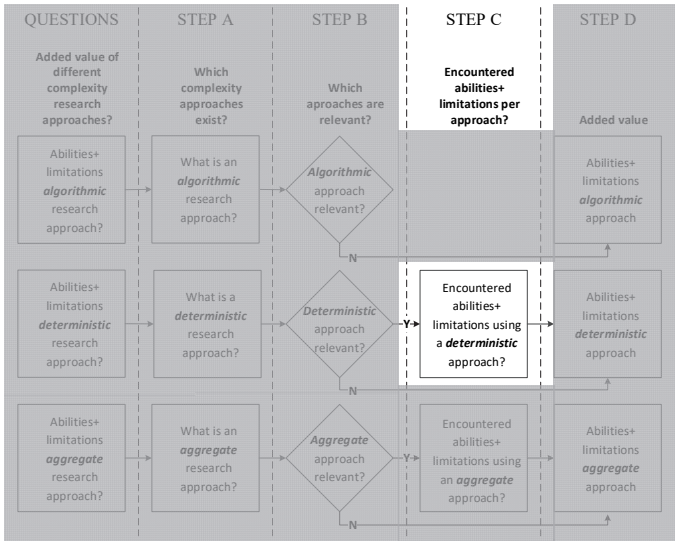
3.6 Conclusions

In this chapter first, we explored complexity as a construct in ERP research. We noticed that although in ERP research often is referred to the complexity of ERP implementation, still only very few authors discuss the meaning of complex and complexity as a construct in their research. Also, these constructs are not discussed specifically related to the ERP implementation process itself. That is to say, what does the complexity of ERP implementation mean? Therefore we tried to answer this question by exploring what do science and practice mean by the construct of complexity and finally by defining what we in our research mean by the construct of the complexity of ERP implementation. However, only a definition of complexity of ERP implementation is not sufficient for exploring which research approaches might be useful to handle this complexity of ERP implementation in research and practice. Therefore we also had to find a suitable differentiation by which we could structure research approaches and map existing and future research for ERP implementation into specific approaches. We encountered Manson's differentiation of research approaches (Manson, 2001) which we consider a clear, practical and well-known differentiation. Manson distinguishes three research approaches into complexity: algorithmic, deterministic and aggregate. We evaluated every paper from our structured literature review to get an indication which of Manson's approaches might prevail in ERP implementation research. A deterministic complexity research approach seems to prevail. Next, we discussed every research approach from Manson to determine the relevancy for our research goal. We concluded that of these three research approaches we considered an algorithmic research approach for our research of too little value. However a deterministic and an aggregate approach seemed both valuable and thus relevant for our research.

Hence in Chapter 4 we will explore the deterministic research approach into the complexity of ERP implementation by discussing design and results of performed own empirical research. Similar in Chapter 5 we will discuss this for the aggregate research approach.

CHAPTER 4: EXPLORATORY RESEARCH INTO DETERMINISTIC COMPLEXITY OF ERP IMPLEMENTATIONS

4.1 Introduction



In the previous chapter, we discussed which of Manson's complexity research approaches we considered relevant for further exploration for our research into ERP implementation complexity. In this chapter, we will explore the **deterministic** complexity research approach into ERP implementations by discussing design, results and conclusions from three performed studies which are implicit of a deterministic complexity research approach.

We will discuss these three studies in detail. These studies contribute to better understanding aspects which are part of the complexity of ERP implementations. Every study we performed has its specific contribution to ERP research, independent of the type complexity research we investigate. Therefore we will discuss relevance, goals, results and conclusions of each study individually.

Moreover, based on our insights we gathered by performing these studies, we will conclude Chapter 3 with a discussion of the abilities and limitations of their implicit deterministic complexity research approach. This discussion will contribute to an-

swering our main research question: "What is the added value of explicit application of different complexity research approaches into ERP implementation?". Therefore this chapter will show results on two levels of abstraction. First, results which contribute individually to mainstream ERP implementation research and second, results which contribute to answering our main research question.

We will present three studies in the form of a research paper. These research papers are included unaltered. Study 1 was presented as a paper at the 6th International Conference on Software and Data Technologies, Seville, Spain in 2011. Study 2 was published in the International Journal of Enterprise Information Systems in 2008, volume 4. Study 3 is submitted as a research paper to the Journal of Computer Information Systems (JCIS) and still under review during the writing of this thesis.

As discussed in Chapter 3 about complexity research approaches, a deterministic complexity research approach does not assume that a phenomenon can fully be understood. Nevertheless, the system has an underlying order, but this order is too complicated to be comprehended and modelled. However, by the discovery of a few key variables, it is assumed that the behaviour of the system can satisfactorily be described and the complexity of its behaviour be understood and handled. With Study 1, 2 and 3 we intended to add value to existing ERP research by expanding knowledge about "variables" which might influence the course of ERP implementations.
In mainstream ERP research, research into critical success factors, risk factors and other influencing factors is extensive. Therefore by *Study 1* we intended to expand this knowledge about factors which affect the complexity. We detected factors which can influence the complexity of the integration process of ERP systems and non-ERP systems. Hence Study 1 aimed at compiling a list of factors which might influence the complexity of the integration process of ERP systems and non-ERP systems. This list of factors can be useful for better judgment of the complexity of this integration process.
In mainstream ERP research, characteristics of ERP implementation project phases are discussed and studied. Different phasings are proposed and studied. The purpose of these phasings is to facilitate planning and control of the ERP implementation project. However, we noticed that on a deeper level a comprehensive collection of ERP activities independent of the various phasings does not exist in ERP research. A comprehensive collection of ERP implementation activities seemed relevant to us, as this might support planning and managing of an ERP implementation project, hence handling its complexity. Therefore *Study 2 and 3* intended to gain a better insight into the complexity of ERP implementations by studying which activities usually are performed within an ERP implementation project. Our study intended to view activities independent of the phase of a project and determined which activities are closely related. Therefore we researched which activity clusters usually exist within ERP pro-

jects. Knowledge of these activity clusters can serve as a model which can support planning and managing of ERP implementations and therefore also reduce the handling of the complexity of ERP implementations.

4.2 Study 1: complexity impact factors on the integration process of ERP and non ERP systems[1]

Abstract

This study shows an expert confirmed initial list of factors which influence the complexity of the integration process of ERP systems and non ERP systems. After a thorough search for complexity factors in scientific literature, a survey amongst eight experts in a leading European long special steel products company, which was recently composed out of independent international companies, was conducted. The participants confirmed the retrieved list from literature, consisting of five quantitative and 21 qualitative factors. The participants added one extra qualitative factor and scored the importance of all factors. Three quantitative factors, i.e. a technology, a business and a project factor, scored highest. When dealing with integration issues, this initial list of factors can provide awareness for organizations to support activities such as planning, control and risk management.

4.2.1 Introduction

Most organizations own a portfolio of many different systems of software modules from different suppliers (Lemahieu, Snoeck, & Michiels, 2003; Light, Holland, & Wills, 2001; Marinos Themistocleous, Irani, & O´Keefe, 2001a), often based on different standards, programming languages and operating systems and unfortunately often insufficiently documented (Marinos Themistocleous et al., 2001a). In many organizations integration of Enterprise Resource Planning (ERP) systems with other systems has been shown to be complex (Sammon & Adam, 2005; Sharif, Irani, & Love, 2005). There are several reasons for this complexity: ERP systems have a monolithic character and initially were not designed to work with other information systems (Klaus et al., 2000). Also ERP systems have limited flexibility because ERP systems force organizations into adapting their business processes according to the ERP system designers view on these business processes (Esteves et al., 2002; Gibson, Holland, & Light, 1999; Koch, 2001). However unique business processes can provide competitive benefits, which forces the organization to use custom build information systems and integrate them with their ERP system. Besides that, there is a growing need for integration between supply chain partners for reasons of cost reduction and coping with the worldwide competition. What's more, instead of using one ERP system for every business process in an organization, there is a tendency for using the best modules of different ERP suppliers. This Best of Breed (B.o.b.) solution prevents insufficient support of the business processes or costly customization (Alshawi, Themistocleous, & Almadani, 2004; Light et al., 2001), but causes additional effort for

[1] This paper originally appeared by authors Janssens, G., Hoeijenbos, M., & Kusters under the title of 'Complexity impact factors on the integration process of ERP and non ERP systems: A basis for an evaluation instrument' in the proceedings of the 6th International Conference on Software and Data Technologies, Seville, Spain, pages 17-22, 2011 (Janssens et al., 2011)

integration of the separate modules. From this we may infer that integrating an ERP system with other systems is an important and complex process in an ERP implementation project.
Understanding factors that influence this complexity should support integration activities. Therefore this paper discusses the results of an initial survey based study into the most important factors which influence the complexity of the implementation process when integrating non ERP systems with ERP systems. First the research goal and approach is described. Second the concept of integration in this research is explained. Third, the complexity factors that influence complexity retrieved from literature are shown. After that, the survey method is discussed, the results from the survey are shown and finally the results, conclusions and further research are discussed.

4.2.2 Research goal and approach

Every information system in an organization has its own basic assumptions and technical basis. Also every system will support different business processes and therefore have a different purpose. Therefore integration of information systems within an organization, with each having their own specific characteristics, will be a complex issue. Because the organizational information systems differ in many areas but nevertheless need to be integrated, it is reasonable to assume that many factors will influence the complexity of this integration process. In the last decades ERP systems have become important information systems which in many cases act as the core or spine of the information systems architecture in an organization (Sharif et al., 2005). In most cases this core is surrounded by non ERP systems that play a vital role within the organization. These non ERP systems often support organizations in providing extra value of services in relation to their competitors or are specific to an organization, and therefore cannot be replaced by an ERP module. Since integration of ERP systems with other systems is important, this research aims at obtaining the factors which influence the complexity of the integration process between ERP systems and non ERP systems.
This was achieved by first performing an extensive literature search for verified factors that influence the complexity of the integration process of ERP systems with non ERP systems. Although several papers mention factors of influence, no research has been found which shows a comprehensive and confirmed list of such factors, and equally important, shows which factors influence most the complexity of the integration process. Therefore a novel list of factors was constructed from research literature in relevant related areas. To be of value for further research and use in ERP projects, such a list must be confirmed by empirical research. In this research a first investigation into the relevancy and completeness of this novel list has been performed by retrieving the opinion of a small group of experts. Experts seemed a pragmatic empirical source for a first confirmation of the retrieved factors, as came clear from literature

that there are many views on the subject and a respectable number of factors should to be taken into account.
The purpose of this empirical research was to retrieve answers to the next questions:

1. Is the list retrieved from literature complete?
2. If not, what factors should be added to the list?
3. What is the relative importance of the factors on this (appended) list?

4.2.3 Integration

In general, integration within the IT community is the creation of links between information systems. Because of existing different interpretations of the word integration, this section will briefly discuss the concept of integration as it is used in this research.
Integration is indicated by different expressions (Marinos Themistocleous & Irani, 2002; Marinos Themistocleous, Irani, & O´Keefe, 2001b): Enterprise Application Integration (EAI), System Integration (SI), Value Chain Integration (VCI), Supply Chain Integration (SCI), Extended Business Integration (EBI), E-Business Integration.
All these expressions point to integration within an organization or integration across the borders of an organization. The purposes of integration for an organization are (Bhatt, 2000): reduce cost, improve services and improve synergy effects. Reducing cost is possible by efficiency improvement by integration of processes and also by reducing the maintenance cost of information systems. The improvement of services results from a faster responsiveness to changes on the market.
Gulledge (2006) states that the term integration is commonly used when discussing enterprise applications. There are several definitions of the term integration such as: 'the interfacing of systems together so they can pass information across a complex technology landscape' (Gulledge, 2006) or 'the extent to which data and application through different communication networks can be shared and accessed for organizational use' (Bhatt, 2000). Unfortunately these definitions are purely oriented towards the technical aspects of integration and leave business processes out of context.
Therefore a more comprehensive paradigm of integration is: ***The integration of data resources, the integration of application functions, and the integration of business processes*** *(Fan, Shi, & Wu, 1999).*
Because in the authors opinion, integration during an ERP project is never just a technical matter, this paper will employ the concept of integration as proposed by Fan et al. (Fan et al., 1999).

4.2.4 Complexity factors in literature

4.2.4.1 Approach

In existing literature no acknowledged list of factors was found that influences the complexity of the integration concept as defined by Fan (1999). For that reason a search in ERP literature and non ERP literature was performed into factors that influence this complexity. By this search 45 relevant papers were retrieved. Only 15 discussed the research topic according to the integration concept by Fan (1999). Systematic evaluation led to the identification of eight main ***areas of influence*** concerning the complexity of integration as shown in table 4.1. These areas of influence are of a too high level of abstraction to be able to be useful as variables for indicating the level of complexity of an integration process. Therefore also ***factors*** within these areas of influence have been retrieved from literature, which are more concrete and if possible can be measured objectively. These factors are also shown in table 4.1.
The definitions, sources of retrieval, motivation why a factor influences the complexity of integration of these main areas and the list of factors with their relative importance are available from the authors.

4.2.4.2 Results

Although literature on the subject of integration is often focused on technical solutions of integration problems and on EAI, this technology focus covers only a part of the integration problem (Sharif, Elliman, Love, & Badii, 2004). More and more suppliers of EAI technology therefore focus on products that make integration of business processes possible (Cakular & Wijngaarden, 2002). EAI not only supports automating activities, but also the improvement and change of business processes (Bhatt, 2000). Business process redesign is an important part of integration.
The quality of the implementation project itself also influences the ease of integration (Thomas H Davenport, 1998; Fui-Hoon Nah, Lee-Shang Lau, & Kuang, 2001 ; Lam, 2005; Sammon & Adam, 2005).
Therefore to be able to indicate complexity factors of integration, this paper discusses them using three logical viewpoints (Klesse, Wortmann, & Schelp, 2005; Lam, 2005; Sharif et al., 2004; Marinos Themistocleous et al., 2001a; Marinos Themistocleous et al., 2001b): ***Technology***, ***Business*** and ***Project***.
As described before, for every influence area factors have been retrieved, see table 4.1. In the current stage of this research it seems more important to retrieve all factors and score them according to their importance, than to do extensive research into defining variables for every factor by which a factor can be measured.

4.2.5 Empirical validation of the complexity factors

4.2.5.1 Introduction

In order to determine the completeness of the list and the relative importance of the factors expert judgement was sought, as this seemed a suitable source of information for this purpose.
The experts had to meet the following profile:'

- Knowledge of the management of the integration process of information systems in general ;
- Knowledge of integration of ERP solutions and complementary IT solutions on at least one but preferable more of the views 'technology', 'business' and 'project';
- To avoid emphasis on the specific circumstanced of a single organization, they should have different organizational (at least) and national (preferably) background.

Experts should add factors if required, rate the relative importance of the factors and react on additional factors and arguments from other experts. A large-scale survey is not suitable whereas in general the availability of experts in this area and willingness to participate is limited. Therefore this research aimed at a small group of experts with sufficient knowledge willing to participate in several rounds in a Delphi setting.

4.2.6 Research Environment

Inviting experts from various international organizations would be the ideal research setting. On the other hand, experts should be available and being able to understand each other's contribution, which pleads for a selection from a single organization. As a compromise we selected experts from an organization that fairly recently was created by a merger between a number of different companies.
The survey was performed amongst experts in a leading European special steel products company. This organization employs 4,300 people at 16 production sites and several sales companies in Europe and the USA. The company is in its present form a young organization, composed of different independent steel companies in various segments of the steel market. Before consolidation the different companies had their own ERP systems, business processes and culture. In this organization there is a major focus on the integration of the different information systems caused by the consolidation of the different units. Recently merged, still variations in organization, business processes and nationality exist. It is reasonable to assume that the outcomes of the survey are of equal value as a survey amongst experts from independent organizations.

4.2.7 Approach

A multi round survey approach through e-mail was in this case a practical tool because of the geographical different locations of the participants. Also, the participants spoke different languages. The survey format allowed them time to understand and formulate their answers in a non-native language. The survey used the following procedure:

1. A first individual rating of the factors and identification of supplementing factors by the participants.
2. Analyzing these additional factors and construction of the final list of factors.
3. Rating of the final list of factors by the participants by the Delphi technique in several rounds until changes in rating per round were minimal.

The survey consisted of predefined questions with predefined answers and in round one an additional open question about supplementary factors.

4.2.8 Participants

The group consisted of five IT Managers, one CFO, one IT software developer and one Information analyst. Also the group was composed of three persons from Finland, one person from Sweden and four from the Netherlands.

Table 4.1 Factors, references and scores

Viewpoint **Areas of influence** ID. Factor	
Technology	
Application Portfolio	
	1. Number of applications
	2. Number of different application types
Choosing the right integration technology	
	3. Extend to which functionality of EAI technologies overlap
	4. Quality of available integration technologies
	5. Number of available and necessary (now and in the future) pre-built adapters
	6. Possibility to develop custom adapters
	7. Availability, within the organization, of a tool to select the necessary EAI technology
Information Management	
	8. Common layer with definitions of objects, relations between objects and business rules
	9. Strategy to handle data redundancy, replication and inconsistence over applications
IT sophistication	
	10. Technical knowledge, within the organization, concerning the current IT infrastructure
	11. Technical knowledge, within the organization, concerning the new EAI technology
Technical Goals to be met	
	12. Level of integration from a technical point of view
	13. Quality of the new IT-infrastructure
27. Availability of new EAI technologies and/or ERP solutions	
Business	
Business Goals	
	14. Level of integration, from a business point of view
	15. Number of organizations that need to be integrated.
	16. End-customer integration.
	17. Business goals to be met
	18. Level of external pressures that are forcing the organization to adopt EAI
	19. Stakeholders goals differ
Knowledge and skilfulness in changing the organization	
	20. Specific EAI knowledge and skills within the organization.
	21. Ability of an organization to adopt new technology.
	22. Employee knowledge of processes
	23. Willingness employees to share control & ownership of processes
Project	
Differences between EAI and traditional IT projects	
	24. Availability of proven EAI methods within the organization
	25. Integration of all existing applications within a portfolio of applications rather than the selection of development of one new application
	26. Number of project owners and stakeholder groups

4.2.9 Results

In the first round, all participants received a questionnaire with the viewpoints, areas of influence and factors. Every factor contained a definition and a reason for inclusion as derived from literature. The participants were invited to ad and motivate factors they missed and rate every factor on a five points Likert scale (very small to very big

influence) along with a motivation. The respondents added one additional factor, number 27 in table 4.1.
In the second round, all participants again received the questionnaire. Besides the views and factors, it contained the summarized motivations from the previous round as well as factor 27. Round 2 did not lead to a major change of opinions. For this reason it was decided to stop the survey and use the results retrieved so far, supported by the announced unwillingness of the participants to participate in a third round. Table 4.2 shows the five top rated factors, while table 4.3 displays the five factors with the lowest scores, both in descending order. The complete list of scores of all factors after round 2 is available through the authors.

Table 4.2 The five top rated factors

Id	Factor
15	Number of organizations that need to be integrated
1	Number of applications.
26	Number of project owners and stakeholder groups
6	Possibility to develop custom adapters.
23	Willingness of employees to share control & ownership of processes.

Table 4.3 The five lowest rated factors

Id	Factor
7	Availability, within the organization, of a tool to select the necessary EAI technology
24	Availability of proven EAI methods within the organization
11	Technical knowledge, within the organization, concerning the new EAI technology
9	Strategy to handle data redundancy, replication and inconsistence over applications
20	Specific EAI knowledge and skills within the organization

4.2.10 Conclusions Study 1

All complexity factors identified from literature are confirmed by the participants in this survey. Although not all are rated equally important, all factors were scored at least a 'small' to 'normal' influence on complexity. However it is also reasonable to assume that the rating will somewhat be influenced by organization specific characteristics. According to the participants of the survey, clearly some factors have more influence on complexity.
The list of the retrieved factors seems rather comprehensive, given that only one factor was added by the participants that was not in the original list and that this factor was rated as having a normal influence on complexity. This fact and that all factors scored as relevant, suggest that the final list is not heavily influenced by the specific

circumstances in this organization. If this would be the case, the authors would expect an explicit variation in the scores.
This survey confirms that integration should not be viewed as a pure technical matter. Three out of the five most important factors are non-technical factors. Two factors have an organizational view and one factor a project view. The list in table 4.1 consists of five quantitative factors (ID's 1, 2, 5, 15 and 26) and 22 qualitative factors. This might suggest that qualitative factors play an important role in the complexity of integration. However the list of the five top rated factors shows three quantitative factors and two qualitative factors. The two most important factors are quantitative factors. Also the list shows that the three quantitative factors belong to the three different viewpoints.
The authors expect that the present list of qualitative and quantitative factors is already a useful instrument for organizations to determine and value the relevant factors which influence the complexity of their integration of ERP with non ERP systems. It can be useful as an instrument for recognition and structured discussion of the important factors which influence the complexity of integration. Usage should provide awareness of the condition of a specific factor in a particular organization in areas like planning, control and risk management.

4.2.11 Discussion Study 1

In this research only the relevant influence factors have been retrieved. Relationships between factors are not discussed although at a glance factors seem related to each other. For instance the three top factors, probably will have a high correlation. For example the higher the number of organisations are, probably the higher the number of applications will be. This aspect should be addressed by further research because this might lead to simplifying the model i.e. reducing or condensing the number of relevant factors.
Of course it would also be very useful for an organization if they could match the factors to the complexity of the integration and match this to the integration effort. Research into finished projects can relate their integration effort to the factors retrieved in this research.
Finally, as the current list is a novel one confirmed by only eight experts in three European countries, more research should be undertaken into the comprehensiveness the areas of interest, the factors and the relative importance of these factors.
Nevertheless the present rated list of qualitative factors can serve as starting point for further research.

4.3 Study 2 and 3

In the next two sections we will discuss two studies which intended to gain better insight into complexity of ERP implementations by studying which activities usually are performed within an ERP implementation project. In Study 2 was a first exploration of ERP activities and clustering of these activities in meaningful clusters. Study 3 was a more extensive study into the same subject, were more experts were involved and also the abstraction level of meaningful clusters was explored. Study 3 complemented Study 2.

4.4 Study 2: Sizing ERP Implementation Projects: An Activity-Based Approach[2]

Abstract

ERP implementation projects affect large parts of an implementing organization and lead to changes in the way an organization performs its tasks. The costs needed for the effort to implement these systems are hard to estimate. Research indicates that the size of an ERP project can be a useful measurement for predicting the effort required to complete an ERP implementation project. However, such a metric does not yet exist. Therefore research should be carried out to find a set of variables which can define the size of an ERP project. The authors hypothesize that ERP projects consist of a collection of clusters of activities with their own focus on implementation costs and project size. This was confirmed in a survey among domain experts. This paper describes a first step in retrieving these clusters. It shows 21 logical clusters of ERP implementation project activities based on 405 ERP implementation project activities retrieved from literature. Logical clusters of ERP project activities can be used in further research to find variables for defining the size of an ERP project.

4.4.1 Introduction

Globalization has put pressure on organizations to perform as efficiently and effectively as possible in order to compete in the market. Structuring their internal processes and making them most efficient by integrated information systems is very important for that reason. In the 1990s organizations started implementing ERP systems in order to replace their legacy systems and improve their business processes. This change is still being implemented. ERP is a key ingredient for gaining competitive advantage, streamlining operations, and having "lean" manufacturing (Mabert, Soni, & Venkataramanan, 2003). A study of Hendricks indicates that research shows some evidence of improvements in profitability after implementing ERP systems (Hendricks, Singhal, & Stratman, 2006). Forecasters predict a growth in the ERP market.

Several researchers also indicate that much research is still being carried out in this area (Botta-Genoulaz, Millet, & Grabot, 2005; C. Møller, Kræmmergaard, & Rikhardsson, 2004). Although the research area is rather clearly defined, many topics still have to be researched and the usefulness of results for actual projects has to be designed.

ERP projects are large and risky projects for organizations, because they affect great parts of the implementing organization and lead to changes in the way the organization performs its tasks. The costs needed for the effort to implement these systems are usually very high and also very hard to estimate. Many cases are documented where

[2] This paper originally appeared by authors Janssens, G., Kusters R., & Heemstra under the title of 'Sizing ERP Implementation Projects: An Activity-Based Approach' in the International Journal of Enterprise Information Systems, pages 23, 2008 (Janssens, Kusters, & Heemstra, 2008a)

the actual required time and costs exceeded the budget, that is to say the estimated costs, many times. There are even cases where ERP implementation projects led to bankruptcy (Holland & Light, 1999; Scott, 1999). Francalanci states that software costs only represent a fraction of the overall cost of ERP projects within the total costs of the implementation project, that is to say, less than 10% over a 5-year period (Francalanci et al., 2001). In addition Willis states that consultants alone, can cost as much as or more than five times the cost of the software (Willis, Willis-Brown, & McMillan, 2001). This is confirmed by von Arb, who indicates that consultancy costs can be two to four times as much as software license costs (Arb, 1997). This indicates that the effort required for implementing an ERP system largely consists of effort-related costs. Von Arb also argues that license and hardware costs are fairly constant and predictable and that only a focus on reducing these effort-related costs is realistic. The conclusion is legitimate that the total effort is the most important and difficult factor to estimate in an ERP implementation project. Therefore the main research of the authors only focuses on the estimation of the total effort required for implementing an ERP system.

In every project there is a great uncertainty at the start, while at the end there is only a minor uncertainty (Meredith & Mantel Jr, 2011). In the planning phase the most important decisions are made that will affect the future of the organization as a whole. As described earlier, a failure to implement an ERP system can seriously affect the health of an organization and even lead to bankruptcy. This means that it would be of great help if a method would exist that could predict the effort required for implementing the ERP system within reasonable boundaries. The method should not be too complex and should be quick. Its outcomes should support the rough estimation of the project and serve as a starting point for the detailed planning in the set-up phase of the project phase and for the first allocation of the resources. Moreover, if conditions greatly change during a project, the method could be used to estimate the consequences for the remaining effort required for implementing the ERP system.
The aim of this paper is to answer which activities exist in ERP projects according to literature and how these can be clustered as a basis for defining the size of an ERP project.
In the paper the approach and main goal of our research will first be described, followed by a literature review on ERP project activities. After that it will present the clustering approach and results followed by conclusions and discussion.

4.4.2 Research approach

When examining more or less successful methods for predicting software development effort, it is to be expected that with regard to implementing ERP systems it will also be possible to find measurements for predicting implementation efforts.
However, Stensrud (2001) already indicated that although many effort prediction systems exist, none unfortunately have been specifically devised for ERP projects.

Kusters and Heemstra (R. Kusters & Heemstra, 2007; R. J. Kusters, Heemstra, & Jonker, 2009) collected candidate cost driver variables from literature and asked experts in two major companies what they thought about the relevance of these variables. One of their conclusions was that the size of an ERP implementation is a major cost driver in ERP implementation projects. In software development the size of the software can be expressed in a single variable such as number of program lines or function points (Stensrud, 2001). By using this variable in a formula with several parameters, useful predictions of the development effort can be made. Can similar variables be found for predicting the implementation effort in an ERP project? According to Stensrud several variables together should be used to express this size. Francalanci (2001) used three variables for her size definition: organizational size, configuration size and technical size. Von Arb (1997) used two variables for size definition in his dissertation: number of users and number of ERP (sub)modules. As far as the authors can conclude from studying available publications on this topic, no further research has been carried out in defining the size of an ERP implementation project. All the mentioned researchers concluded that size cannot be expressed as a single variable as in software development, but should be expressed as a multidimensional variable. ERP implementation projects are complex projects where successful organizational, technical and people strategies are critical factors for success (Aladwani, 2001; Ngai et al., 2008). Because an ERP implementation project is confronted with many different aspects, the authors postulate the hypothesis that an ERP implementation project consists of a collection of clusters of activities with their own focus on implementation costs and project size. Clusters of activities include: the preparation of the appropriate technical infrastructure, the business process redesign or the installation of the software. Of course these clusters of activities will be related to each other, but the authors expect that they will influence the total cost of the implementation project fairly independently. If size variables can be found for these clusters and these variables could be used as an estimator for the prediction of the effort required for these clusters, these variables could be the dimensions of the multidimensional variable which defines the size of an ERP implementation project.

For the development of regular information systems several methodologies exist, which support the project manager in deciding what needs to be done in the project. In these methodologies, all relevant activities are described and defined in terms of goals, results and necessary resources. In software development projects, activities that are relevant in that specific situation are selected from this methodology and planned. It goes without saying that not all activities are relevant in every project. There is no reason to expect that an ERP implementation project will be different in that matter. Therefore this research is based on the assumption that a range of activities exists which represents the most relevant activities in an ERP project.
The author's research approach in finding the most relevant activities in ERP implementation projects is to retrieve them from published research. Although several au-

thors showed the phases in an ERP project and activities in these phases (Robey, Ross, & Boudreau, 2002), a complete list of all relevant activities in an ERP implementation project was not found, unfortunately. Several authors pointed out activities which where relevant according to their point of view in their paper, but none of them intended to collect all possible relevant activities. Therefore papers were collected which listed activities within an ERP implementation project. By examining papers with different views the authors of this paper expect to have found the most relevant activities.
In this paper the authors try to lay a foundation for defining the size of an ERP project. Because it is expected that the costs for the effort to implement an ERP system will constitute the greatest part of the total costs of an ERP implementation project, the first logical step is to define which activities that require human effort are important in an ERP project. Activities are always performed for a reason, i.e. to reach a certain goal and can be grouped into logical clusters which contribute to the same intermediary product or products. For instance, an intermediary product such as 'trained users' can be achieved by a cluster of activities such as: 'prepare training material', 'train the trainers', 'set up training infrastructure', 'train users' etcetera.

4.4.3 Objective of this research

The objective of this research is to define logical clusters of ERP project activities. This paper will show the method and results in retrieving important ERP activities and the results of this first formal attempt to cluster these activities into clusters which contribute to similar intermediate products. This paper aims at answering the next research questions:

1. Is the hypothesis of the authors, that ERP projects consist of a collection of clusters of activities with their own focus on implementation costs and project size, supported by domain experts?
2. Which activities in general exist in ERP projects according to literature?
3. What is a useful method to cluster these activities?
4. What is the result of a first clustering of these activities?

Is the hypothesis of the authors, that ERP projects consist of a collection of clusters of activities with their own focus on implementation costs and project size, supported by domain experts?

The main research question of the authors focuses on the estimation of the effort-related costs needed for the implementation project of an ERP system. However, is it a relevant research question? That is to say, are the results of this research relevant in the empirical world? In order to detect the relevancy, the authors executed a small survey in the period September until November 2007. In this survey they also checked

the reasonableness of their assumptions on a model for estimation of effort-related costs.

Goals Of The Survey

With this survey the authors aimed at retrieving a clear answer from professionals with sufficient knowledge and experience in ERP implementation projects for the following questions:

1. Is it difficult to estimate the total costs of an ERP implementation project?
2. Is it important to estimate the total costs of an ERP implementation project?
3. Could a model support the estimation of the total costs of an ERP implementation project?
4. If there existed a model with clusters of activities and it would be possible to estimate the effort-related costs per cluster, would this be a basis for estimating the effort-related costs for an entire ERP implementation project?
5. If this model would be useful, out of how many clusters would it exist and how many of these clusters would cause the largest part of the effort-related costs?

Survey Content

The survey consisted of two groups of Dutch questions and general instructions.
The first group of questions aimed at retrieving information about the authors' main research questions i.e. whether it is useful doing research at methods for being able to predict the effort-related costs in an ERP implementation project. The purpose of the second group of questions was to verify the assumption that the total cost of the human labour could be predicted by adding the estimated costs from every activity cluster. Also in this group the respondents were asked to estimate out of how many clusters of activities an ERP project consists and how many of these clusters would cause the largest part of the total cost.
Finally the respondents were asked whether they would be willing to participate in other similar surveys.

Target Group Of Participants And Survey Tool

For this survey the authors aimed at a group of professionals with substantial experience, knowledge of and insight in ERP implementation projects.
Because SAP is a commonly used ERP software and is used by big and complex organizations, individuals with experience and knowledge of SAP implementation projects were suitable participants for this survey. Therefore this survey was submitted at a small Dutch conference on the subject of the costs of the maintenance of SAP implementations. The participants could be expected to fit the requirements.
The respondents all attended the conference. The authors chose to use an online survey as their research tool. Mainly because they expected that more participants would

respond to an online survey than to a paper based survey submitted during or at the end of the conference. LimeSurvey was selected as the online survey tool. LimeSurvey (http://www.limesurvey.org/) is an open source survey tool under GNU General Public License.

Survey Results
Initially 42 e-mails were sent to the participants of the conference. From these 42 participants two replied that according to their view they had insufficient knowledge for completing the survey and would therefore not participate. From the remaining 40 participants 20 finished the survey in the period September 13, 2007 until November 7, 2007.

Survey Conclusion And Discussion
The respondents provided unambiguous answers to the research questions:

1. It is rather difficult to estimate the total effort-related costs of an ERP implementation project. (65 %)
2. It is important to estimate the total effort-related costs of an ERP implementation project. (85 %)
3. A model could support the estimation of the total effort-related costs of an ERP implementation project. (90 %)
4. A model with clusters of activities that can be used to estimate the effort-related costs per cluster, can be a basis for estimating the total effort-related costs for an ERP implementation project. (65 %)
5. There are around 10 clusters of activities in an ERP project.
6. From these clusters less than half account for the largest part of the effort-related costs.

This indicates that the main research question of the authors is useful and also that the research direction, by defining activity clusters, is plausible.

The authors are fully aware that the number of respondents is low. Of course a larger group could make the outcomes more reliable. On the other hand, the quality of the respondents is also an important factor. Since the conference was by invitation only, it provided a good quality filter for the participants. On this basis we believe we can have confidence in the results of this survey.

A discussion of this survey in more detail is available as a working paper (Janssens, Kusters, & Heemstra, 2008b).

4.4.4 Which activities in general exist in ERP projects according to literature?

Literature Review On ERP Project Activities

A literature search was performed aiming at finding papers in which activities within an ERP implementation project were listed. From these papers a collection of names and expressions of activities was retrieved. The papers were retrieved from a collection of about 200 papers which were composed of papers selected from 'A Comprehensive ERP bibliography - 2000-2004' from Møller et al. (2004) and a separate literature search for papers about implementation projects of ERP systems. Within this collection of about 200 papers a paper was selected if it showed at least one list of activities performed in ERP selection, implementation or maintenance. In most cases papers were found that enumerated the important activities in a regular project phase of an ERP implementation project. A total of 24 papers were found with lists of ERP activities. These papers can be divided into three categories:

A. Papers which relate risk factors and Critical Success Factors (CSF's) or other influencing factors to activities and/or project phases.
B. Papers about cases which describe the phases and activities of the actual projects.
C. Papers which describe standard project phases and activities from consultancy firms or ERP software suppliers.

It can be expected that these three types of papers will show the important project activities.

Appendix 4.1 shows the list of the retrieved papers and the classification into the three categories.

The next section will discuss the retrieved papers grouped by the three categories. Although the authors aimed at activities that are part of the implementation project, activities were also recorded in this literature study that belong to the pre-implementation phase and maintenance phase of an ERP system.

Papers with research-based phases and activities

These research studies relate risk factors, critical success factors or other influencing factors to activities and/or project phases. These authors based their framework of the standard activities and project phases on other scientific research and in some cases performed interviews with experts to enhance their framework.

A first example of this type of research is by Parr and Shanks (2000). The purpose of their research was to create a project phase model (PPM) of ERP project implementa-

tion. They based their model on other process models of ERP implementation from other researchers and tried to synthesize these models into one model which also recognizes the importance of the planning and post-implementation stages. They used the model in two case studies to examine the relationship between the CSF's from their earlier research and the phases to the PPM.

Rajogopal (2002) used a stage model to analyse six manufacturing firms that had one of the widely used ERP systems to retrieve factors of influence in the various stages of ERP implementation. He based his stage model on a six-stage model from Kwon and Zmud (1987) and other authors.

Al-Mashari et al. (2003) presented a novel taxonomy of the critical success factors in the ERP implementation process. They based their taxonomy on a comprehensive analysis of ERP literature combining research studies and organizational experiences. In their taxonomy they showed three major ERP phases. In these phases they also described project activities based on an analysis of ERP literature.

Ehie and Madsen (2005) studied 38 critical issues in ERP implementation to measure the critical factors of ERP implementation. They developed a questionnaire based on five stages of ERP implementation. Stages are based on reviews of literature and extensive personal interviews with ERP consultants.

In their investigation on critical management issues in ERP implementation Kumar et al. (2003) divided the project activities into two phases 'dollars to assets' and 'assets to impacts'. They described the typical activities within these phases. They based their phase and activities on innovation process stage models from other authors. They used these activities in open-ended questions in a questionnaire for ERP project managers of 20 Canadian organizations. The aim of the questionnaire was to find critical management issues.

Hallikainen et al. (2006) developed and tested a model to support the decision which modules are implemented and in which order. They based their model on the phase model of Bancroft.

In their paper in which they seek to provide a conceptual model that explains the complexity of an ERP system to project managers in a non-technical manner, Marnewick and Labuschagne (2005) also present an ERP implementation methodology, which consists of five steps.

Somers and Nelson (2004) examined the ERP project from different viewpoints: Players, ERP Project Life Cycle Stages and Activities. Their main purpose was to analyse the importance of key players and activities across the ERP life cycle by designing a questionnaire, which was returned by 116 companies. They adopted the six-stage model from Rajagopal (2002). For every phase they derived the key activities from other research studies.

The same six-stage model was used by Somers and Nelson (2001). They questioned 86 organizations in order to retrieve the impact of Critical Success Factors (CSF's) across the stages of ERP implementations. The top CSF's that were listed for every ERP implementation stage largely consist of project activities.

Umble et al. (2003) identified CSF's, software selection steps and implementation procedures critical to a successful implementation. Based on available resources and own experiences, including a case study, they showed the most important activities for ERP system selection and implementation steps.
The activities for selecting an ERP system were presented by Wei and Wang (2004). They constructed a comprehensive framework for selecting an ERP system and applied it to a case in Taiwan. This was followed by a research paper in which they presented a comprehensive framework for selecting a suitable ERP system, which was based on the analytic hierarchy process (AHP) method from Saaty (Wei et al., 2005).
Wagner and Antonucci (2004) studied whether there are different ERP implementation approaches and models for a large-scale integrated ERP system in the public sector as compared to the private sector. For their research they used a generalized structured implementation.
Markus and Tanis (2003) described various subjects of ERP systems for educational purposes. They based their phases on other models from other authors. For every phase they described typical activities, common errors or problems, typical performance metrics and possible outcomes.
Latvanen and Ruusunen (2001) used a socio-technical model of risk management of ERP projects.
Mabert et al. (2005) compared and evaluated the use of regression analysis, logistic (logit) models, discriminate analysis and data envelopment analysis (DEA), for empirical data from ad surveys of ERP implementations in the US manufacturing sector. For this they applied key planning, decision and implementation management variables for the implementation phases. They did not specify important activities within these phases.
Sumner (2000) identified risk factors unique to ERP projects by interviewing ERP project managers in seven companies. For this research she used five ERP project phases.
Francalanci (2001) tested whether technical size and organizational complexity of SAP/R3 projects could be used to predict the implementation effort. She used ERP implementation phases that were consistent with the reference manuals of most commercial packages.
Weston (2001) discussed project management issues related to four ERP implementation stages.
Esteves and Pastor (2001) analysed the relevance of critical success factors along SAP implementation phases. They used the five implementation phases from the ASAP implementation methodology.

Papers with case-based phases and activities
These research studies present case studies of ERP implementation projects. The purpose of these studies is to show in detail what happened in an actual case or to use a case to test a construct.

Berchet and Habchi (2005) studied an ERP implementation project at Alcatel. The project was carried out according to a five-stage model. They also described important activities for every phase.
In describing the ERP implementation at Rolls-Royce, Yusuf et al. (2004) carried out an in-depth study of the issues behind the process of implementation. The implementation plan at Rolls-Royces consisted of four main phases. In their description of these phases the main activities were also described.
Sarker and Lee (2003) tested three critical success factors in a case. They concluded that only the CSF 'strong and committed leadership' could be empirically established as a necessary condition. The case company implemented ERP according to three phases.
Tchokogué et al. (2005) performed a case study and showed the lessons learned in that organization at a strategic, tactical and operational level. The project studied had five phases.

Papers with Project phases from consultancy firms and ERP suppliers
One paper specifically described ERP implementation methodologies used by consultancy firms or ERP suppliers.

Bruges (2002) showed the phases and main activities from three methodologies: AcceleratedSAP (ASAP), The Total Solution (Ernest & Young) and The Fast Track Workplan (Deloitte & Touche).

Retrieve activities
The list of activities was retrieved from these three types of papers. Because the intention is to cluster these activities into logical units, no attention was paid to the phases mentioned in the papers. As shown above there is a variety of the numbers and names for project phases. Therefore only the activity names were retrieved.

With regard to every ERP activity that was discovered, the following was recorded: the paper title, the name of the ERP phase as mentioned in the paper (if present), and the name of the ERP activity itself.
In total 402 activities were recorded. Of course the same activity was mentioned more than once. Double names, synonyms or homonyms were not filtered out for reasons as discussed below in the metaplan session. These activities should be categorized unbiased. A filtering of the activities before the session would result in activities that would be selected and named by the personal preference of the researchers.

4.4.5 What is a useful method to cluster these activities?

A grouping technique was needed in order to be able to categorize the retrieved activities into coherent clusters of activities. As mentioned before, the selection and testing of the clustering technique was also a research goal.
The only categorization found in literature was grouping of activities by formal project phases. Unfortunately, there is no generally accepted phasing for ERP implementation projects. Besides, although activities may be started in a particular project phase, activities can still go on during other phases of the project. Project phases are based on a time-based view of the project. In the concept behind this research the time-based view is not relevant, only what has to be done in the project.

Categorization of project activities by applying objective attributes of these activities, for instance the duration of an activity, was also not possible. Except for its name and in most cases the project phase name, no more properties of an activity were available. However, people with sufficient knowledge of ERP projects should understand an activity. Therefore the clustering can only be done by human judgment.
The number of established activities (402) also implies the need for a formal technique. For this type of clustering a card sorting technique seems appropriate. Card sorting has proven its usefulness in many concept mapping studies (Trochim, 1989). If card sorting is done by one human individual, bias and limited knowledge will influence the result. Judgment by several individuals and group interaction will improve the quality of the results. Unfortunately members of freely interactive groups are often dissatisfied with group interaction (Howard, 1994). According to Howard, a Nominal Group Technique (NGT) improves the output and satisfaction of the group members (Howard, 1994). Therefore, the metaplan technique for the clustering was chosen in this phase of the research. The metaplan technique uses card sorting and can be viewed as a Nominal Group Technique (NGT). The metaplan technique was developed by Wolfgang and Eberhard Schnelle. It is a simple visual technique which can be used by groups to structure thinking processes within the context of group work. A moderator leads the group discussion. Ideas are generated by group members and noted on cards. Finally, these cards are organized into categories and may show new results of which the single persons were not aware. The moderator leads the organizing into categories.
Metaplan is a technique in which cards are sorted by a group of people in a formal way. There is a formal interaction within the group with regard to the categorization. Moreover, sorting of a large number of cards can be done in a relatively short period of time. Last but not least, a metaplan session is easy to setup and requires only a few resources. By using this method, the authors could quickly see whether card sorting by a group would be a useful tool for the clustering. In the next step of the main research the most appropriate method and tool for the clustering should be selected.

This metaplan session was performed as a first step in categorizing i.e. clustering ERP activities in clusters which are logical groups of activities in an ERP implementation project which contribute to the production of the same intermediary products. Of course the activities found in the papers are not comprehensive. However, it is reasonable to expect that the activities mentioned in these papers are important activities in an ERP implementation project and will influence the total project effort. Furthermore, it is not the purpose of this research to find all possible activities. This research tried to find only important activities because they will influence the total project effort most likely. The goal of this first session was to find out whether activities can easily be clustered and if a technique such as the metaplan technique can be used in future to improve the clustering by more experts.

The first step in a regular metaplan session is a brainstorming part from which ideas are generated and noted on cards. In this case there was no brainstorming session for retrieving possible ERP activities. This was replaced by retrieving activities from relevant scientific papers in which phases and activities within these phases were described. The list retrieved from these activities is probably more complete and relevant than by brainstorming. Of course there are many synonyms and homonyms, but this also will be the case in an actual brainstorming session. Only the categorizing part of the metaplan technique was used. Of these activities the following data were printed: name, project phase (if present) and title of the paper. Some examples of these stickers are shown in appendix 4.2. Not only was the name printed on these stickers, because if the name itself would be confusing, it would be possible in the metaplan session to retrieve the paper from which the activity originated to obtain some clarification. The stickers were stuck to 402 Post-it notes which were used in the metaplan session.
The metaplan session was performed by the authors of this paper in a 3-hour meeting. The session was prepared by the first author who selected the useful papers and recorded the activity names, project phases and paper names in an Excel spreadsheet. From these data the stickers were printed and stuck to post-it notes.
The participants of this session were instructed to categorize these post-it notes into logical clusters by sticking them on a wall. The participants had to categorize these notes by bearing strongly in mind that clusters should not relate to project phases, but that activities within a cluster should strongly contribute to the same intermediate product or products of an ERP implementation. After assigning all relevant activities to a cluster, the clusters were studied by the group in detail, which resulted in some rearranging of activities and also in some subgroups within the main clusters.

In this session the first author of this paper served as a facilitator/moderator by taking a Post-it note, reading aloud the name of the activity. After that, the group decided under which cluster of activities the activity belonged. If a cluster did not exist yet, the name of the cluster was mutually decided upon and written on a blank Post-it note.

This was stuck to the wall and the activities belonging to this cluster were stuck below. If an activity resulted in a new cluster, some already categorized activities were, if necessary, moved to this cluster. Some activities were regarded as not being part of the scope of an ERP project or very confusing to the group. They were stuck on a wall separately in an 'out of scope' section. Near the end of the session all relevant activities were assigned to a cluster. After that the clusters were studied by the group in detail, which led to some rearranging of the activities and also to some subgroups within the main clusters. After the session the clusters and activities in these clusters were recorded in a spreadsheet. In addition, obvious double activities and synonyms were removed in a two hour separate session by the first two of the authors. In this session also the cluster names and logical sequence were enhanced.

From the outcomes of the session it can be concluded that the metaplan technique is a suitable technique for clustering ERP activities. The activities taken from literature were categorized according to their name. In the papers there was often no more information available about the exact content of the activity. Therefore in some cases the metaplan group had to further discuss the activity.

Preparing the session was a labour-intensive process. The session itself took about three hours, mainly caused by the large number of activities (402). The categorizing itself was not a difficult task. Sometimes there hardly was any discussion about the naming of the clusters and the assignment of the activities to the clusters. The method could also be useful in subsequent research where other experts should perform the same exercise. Although for practical reasons it would be advisable to perform this session by applying a method and software to do the clustering independent from time and place. Experts are hard to persuade to participate in these sessions. If experts could perform the clustering whenever they want and wherever they want, the willingness to participate will be higher. As shown by Howard as well, support of this process by a Group Decision Support System (GDSS), which can support clustering in different locations and/or at different times, leads to the same quality of results (Howard, 1994). Therefore the authors will try to set up a GDSS for this purpose in the next step.

4.4.6 What is the result of a first clustering of these activities?

Table 4.4 shows the found clusters and sub clusters.
Appendix 4.3 shows all results, i.e. the clusters and sub clusters with all activities and the references from which the activity was derived for every activity.

Table 4.4 also shows that 208 unique activities were assigned to the clusters and/or sub clusters. In the second session the homonyms and synonyms were removed, which resulted in 208 unique activities.

In the second session the clusters were also categorized in three groups: 'project', 'system', 'organization', as shown in Table 4.4. The group 'project' shows the clusters which contain activities required for the proper operation of the project, for instance the project management. The group 'system' shows clusters of activities required for the configuration an implementation of the ERP system itself. Finally the group 'organization' shows clusters of activities required for the organizational changer for the implementation. These points of view can be used in future research for crosschecking whether all relevant activities and clusters are taken into account.

Table 4.4 Found clusters and sub clusters

		Group view			
Clusters	**Sub clusters**	**Project**	**System**	**Organization**	**Number of unique activities**
Selection	Vendor selection		✓		4
	Product selection		✓		16
Project configuration		✓			19
Project management	Management	✓			4
	Communication to organization	✓			4
Organizational and system design	Current state analysis			✓	5
	Organizational requirements			✓	7
	Requirements ERP system		✓	✓	8
	High level Design		✓	✓	6
Configuration and installation	System configuration		✓		17
	Data conversion		✓		4
	System integration		✓		9
	ERP system testing		✓		14
Customizing			✓		7
Infrastructure			✓		14
Reorganization				✓	11
System implementation				✓	21
Training	Training Implementation Staff	✓			2
	Training users			✓	9
	Training maintenance staff			✓	2
Set up maintenance			✓		25
TOTAL					**208**

4.4.7 Conclusion and discussion Study 2

The small survey among ERP experts confirmed the hypothesis of the authors that ERP projects consist of a collection of clusters of activities with their own focus on implementation costs and project size. It also gave a first indication of the number of clusters. Research into defining clusters of activities of ERP projects is therefore relevant and if the estimated number of clusters by these experts is reasonable, this

number can lead to a practical prediction method. The clustering of the authors took place before the survey, therefore the authors where not biased by the outcome of the survey. It is remarkable that the number of activity clusters fairly corresponds with the estimated number of clusters by the correspondents of the survey. A number of approximately 10 clusters seems reasonable.
The most important results of the research described in this paper are clusters of activities. It forms a basis for further research on this subject. The clustering has been done by the three authors. Although this is a small group and they share collective opinions, the clustering has been done in a sound manner and the results are a good indication of what the final result of clustering could be. It will be used as a starting point for further clustering.
The results will be validated in future research by increasing the clustering group of people. Validation will also take place by checking these activities against activities retrieved from real- life projects and checking whether activities from real-life projects can be categorized according to the established clusters of activities. It should of course also be checked whether the activities that can be found in real-life project documentation occur in the list of activities from the literature search.
As described before, the metaplan technique was found in principle to be a suitable technique for clustering these activities. Unfortunately, the preparation is very time consuming and it will also be difficult to arrange this type of session with several experts in this field of knowledge. The use of a GDSS (Group Decision Support System) can facilitate this. It will also have to be researched which GDSS will be the most appropriate and what type of NGT (Nominal Group Technique) should be used. Online open card sorting combined with Delphi technique characteristics could be an option (Paul, 2008).

The results of this paper will be used to perform a first exploration into the practical use of the clusters for defining variables which could be used to define the size of an ERP implementation project. As discussed in the research approach, the size of an ERP implementation project should be expressed in a multidimensional variable. At this point in time the authors assume that the clusters can serve as the dimensions according to which an ERP implementation project can be viewed. Validated clusters are homogeneous groups of activities which can facilitate estimation of the important parts of an ERP implementation project.
The first impression of the authors is that the sub clusters and not the clusters should be the starting point for the definition of variables, because the level of detail of the clusters seems to be too low to be able to easily find variables. However, this has to be verified in further research.

4.5 Study 3: an expert based taxonomy of ERP implementation activities[3]

Abstract

ERP implementation projects are complex and expensive projects. Generally, the complexity is managed by splitting the project into phases. However, splitting the project into phases seems not to enhance the understanding of the underlying processes. Therefore this research aims at enhancing the understanding of these underlying processes through an expert based taxonomy of implementation activities, independent of time and phasing. This taxonomy has been developed by retrieval of 205 ERP implementation activities from literature, a grouping of these activities by 11 ERP implementation experts and comparison with a previous similar study. The method used for grouping was Delphi card sorting which was supported by Websort as a web-based card sorting tool. The proposed taxonomy can serve as a base for further research into ERP implementation projects and can support the management of ERP projects.

4.5.1 Introduction

ERP implementation projects are complex projects (Toni M. Somers & Ragowsky, 2003). This complexity is confirmed by the results of numerous case studies and also by research studies into risk factors for ERP implementation projects (D. Aloini, Dulmin, & Mininno, 2007). In general, the complexity of an ERP implementation project is managed by introducing phases or stages in an implementation project, which will enhance the overview in time (Rajagopal, 2002). However this type of control appears to be insufficient since often the projects are too late, over budget, not embraced by the users or don't realise the expected benefits (Wong, Scarbrough, Chau, & Davison, 2005).

One reason for the inadequate control of ERP projects might be that managing the project by mainly focusing on phases is insufficient as phases enhance the overview in time, but not the understanding of the underlying process (Robey et al., 2002). Therefore this research aims at enhancing the understanding of this underlying process by introducing an expert based taxonomy for implementation activities, which is independent of time and phasing. This expert based taxonomy can be used to add a new perspective to research into and concrete management of ERP implementation projects. For instance, this taxonomy can serve as a starting point for definition and management of subprojects.

This paper describes the methods and results for the design of this expert based taxonomy. A collection of activities was formed by retrieval of activities which occur in

[3] This paper is submitted to the Journal of Computer Information Systems (JCIS) by authors Janssens, G, van der Velde-van Moorst, L. & Kusters, R. and title 'An expert based taxonomy of ERP implementation activities'

the literature concerning ERP implementations. Eleven experts grouped this collection into coherent collections of activities which in turn form the basis for the taxonomy. First, we will explain why this research is relevant. Next how activities referenced in existing literature which discussed the implementation of ERP systems, have been extracted and cleaned up on homonyms and synonyms. After that, we will discuss how a grouping method, an online supportive tool for this method and a group of experts have been selected. Next, we will discuss the results of the grouping by the experts and the comparison of these results with previous research (Janssens, Kusters, & Heemstra, 2010). Finally, we will draw conclusions from the results of this research and the comparison with our previous research and propose a first taxonomy.

4.5.2 Background

One option to manage the complexity of an ERP implementation is by having thorough understanding and control of the most important implementation issues. Therefore it is relevant to be aware of the most critical factors for controlling the implementation. Extensive research has been done into these critical success factors for implementing ERP systems. Huang reviewed work published in various journals and special conferences on the topic of Critical Success Factors (CSF) of Enterprise Resource Planning (ERP) system implementation between 1998 and 2007 (Huang, 2010). Table 4.5 shows the top ten critical success factors he retrieved.

Table 4.5 Top ten ERP implementation Critical Success Factors (Huang, 2010)

1. Top Manager Commitment
2. Teamwork and Composition
3. Education and Training
4. Project Management
5. Definition of Scope and Goals
6. Business Process Redesign
7. Change Management Program and Culture
8. Champion
9. Open and Honest Communication
10. Choose the Right Vendor Right Package

In most cases implementing an ERP system results in the execution of a large project. Therefore, it is not surprising that this overview contains several critical success factors belonging to project management topics: 'Teamwork and Composition', 'Education and Training', 'Project Management', 'Definition of Scope and Goals', 'Champion'.
To effectively manage an ERP project it is custom to split a project into meaningful phases, which determine in what order the activities should be undertaken to reach

an intermediate or end goal of the project. Research into ERP implementation aspects also strongly focuses on these various phases in an ERP implementation project. Rajagopal (2002) collected several phase models and argued that the model from Kwon and Zmud (1987) seemed appropriate, which Somers (2004) supported. The phases are very similar and are often only distinguished by the amount of detail. These phasings serve the project manager well by cutting the complex total implementation into less complex parts which the project manager can manage separately and which are easier to overlook. However, these phases are mainly focused on time and sequence of all project activities and less on the understanding of the activities and purposes themselves. Also, every phase consists of a large amount of activities, which in some cases have no or only minimal mutual relationships. For instance training of users will have only a minimal relationship with the implementation of the technical infrastructure. The main purpose of phasing is to cut the project in time into smaller and better manageable parts.
However cutting the project into phases will reduce the complexity by enhancing the overview in time, but will not reduce the complexity by improving the understanding of the actual processes needed to embed the ERP system into the organisation. That the phases do not improve understanding of the actual processes is supported by Robey et al. (2002), who states that "*Stage theories allow participants to anticipate future challenges, but they do not provide an understanding of the underlying process*".

Division of ERP implementation in phases which are individually managed (Robey et al., 2002) ignore the fact that activities and hence the intermediate products are often performed through the phases. In the example of user training, in the initiation phase, it will be determined how many users in the organisation must be trained. In the implementation phase of the system itself, the training will be designed and will take place, and finally in the phase following the implementation additional training and support must be offered to users. The training dimension is thus relevant through the entire project cycle and does not belong to one phase alone, although the majority of the effort for training in most cases will be needed in the adaptation phase. Figure 4.1 shows the difference between the phase viewpoint and the viewpoint on various meaningful collections (illustrated by some examples) independent of phases.

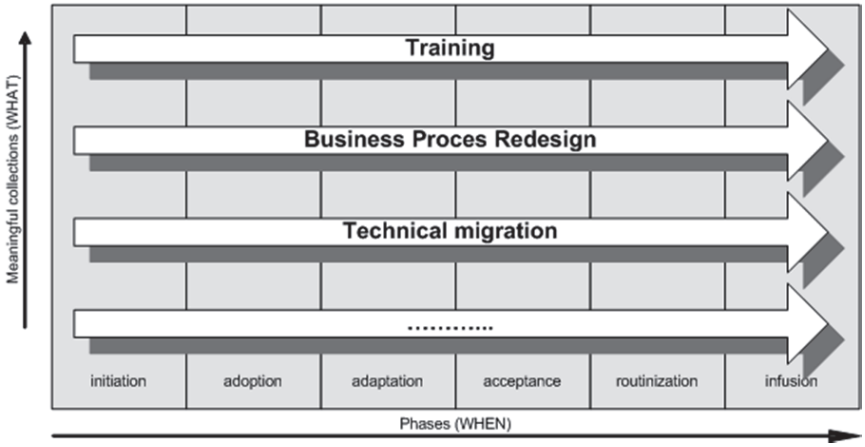


Figure 4.1 Phases of ERP implementation projects versus meaningful process collections

Our research tries to enhance the understanding of these underlying processes in ERP implementation projects by retrieving meaningful collections of coherent activities or processes, independent of phases and viewed over the entire project. The premises behind this is that by better understanding what coherent activities are needed to realise the project, project management will improve because of the improved understanding of the underlying processes thus reducing the complexity of managing the ERP implementation.
In a small survey among ERP implementation experts, we inquired whether these meaningful collections could decrease the complexity (Janssens et al., 2010) of managing the project. We also asked for an indication of the number of collections and whether such collections of activities could be identified. In the expert's opinion in that survey, it would be possible to design these collections. The experts also re-

sponded that these collections would reduce complexity, and eight to 10 meaningful collections of activities might be reasonable.
To retrieve these collections of coherent activities, we intend to form collections of ERP project activities which have a common purpose in the implementation route. For instance, a common purpose might be to reach a state that users become sufficiently trained for the ERP system or a state that the ERP software is implemented and up and running.
The current research seeks to retrieve the various collections of activities of an ERP implementation, but more concrete: *an expert based taxonomy of ERP implementation activities*. The premise behind this is that if it is possible to determine these collections in an unambiguous way, it will enhance the understanding of the implementation processes in projects and therefore in time can reduce the complexity of ERP implementation projects in general. Also, the taxonomy can be used to support project managers in the design of the necessary subprojects for the ERP implementation project. Besides that, the collections of activities can be used for research and practical application in various domains of ERP implementation topics: for example planning, management of implementation, stakeholder analysis, communication with consultants and ERP suppliers et cetera. The hypothesis is that these collections can be retrieved by determining which groups of activities in an ERP implementation are strongly interrelated throughout the complete project. In this context, strong interrelations mean that these activities focus on the same intermediate product within the project. For example, a product could be a sufficiently trained group of users
The aim of this research is to determine an expert based taxonomy of activities which represents the implementation of ERP systems, regardless of any formal phasing, to increase the understanding and management of ERP implementations.

4.5.3 Research approach

An important consideration in designing a new taxonomy is its required level of abstraction. Obviously, the abstraction level is dependent on its purpose. This research aims at a rather high level of abstraction since the taxonomy does not yet exist and also the purpose of this research is to form an initial base for further research. Furthermore based on our survey (Janssens et al., 2008b) and previous research (Janssens et al., 2010) a taxonomy of 8 – 20 categories would be an appropriate level of abstraction. On the other hand, it can be expected that it is impossible to determine an unambiguous level of abstraction as various purposes require various levels of abstraction. These various levels of abstraction are for example also the case for the stage theories mentioned before (Rajagopal, 2002), which are very similar but vary in detail i.e. level of abstraction.
Although in our research experts designed the taxonomy, we could not specify to the experts what level of abstraction was needed other than by indicating in the expert's instructions the required level by examples of intermediate and end products of an

ERP implementation. Therefore, we were also interested in the levels of abstraction which experts would find relevant i.e. according to which level of abstraction the experts would group the activities. Besides that, we were interested in whether experts would reach consensus about the level of abstraction or whether this would be an issue in the grouping process.
Even though this research classifies 205 activities into groups, its purpose is not to create the ultimate correct allocation of every activity to a group. The purpose of this research is to find the main stable cores of activities by which the final groups derive their legitimacy. If different experts allocated some activities to different groups, i.e. the experts did not come to an agreement; this would not influence the outcome. Of course, this should only be the case for a small percentage of the total number of activities. This small number of arguable activities can be considered as 'noise' which should be ignored in the forming process of the taxonomy. This research therefore also intends to explore whether experts can reach consensus about a taxonomy of ERP implementation activities
First, these ERP implementation activities had to be retrieved to enable grouping of ERP implementation activities to form a taxonomy. Since we did not encounter a comprehensive collection of these activities in literature, we extracted a collection of from the literature concerning the implementation of ERP systems. The premise is that the activities which appear in literature must be of significance in ERP implementation projects. Of course, it is expected that not all possible ERP implementation activities can be retrieved in this way. However, the goal of this research is not to retrieve the ultimate collection of ERP implementation activities, but merely the most important/prevailing ones which can be a basis for defining the coherent groups of activities in ERP implementations: the taxonomy.
Second, the retrieved activities had to be refined, as activities can appear in the literature as synonyms, homonyms and can have different wording. For example 'training', 'education', 'user training', 'training of users' or 'train the users'. This refining had to be done in a controlled manner and could only be done by human judgment.
Third, these refined activities had to be grouped into meaningful collections. In this research, experts were chosen as an information source for these collections. The experts used a card sorting method for this purpose.
In the next sections, the three steps are further elaborated.

4.5.4 Collection of ERP implementation activities

One author performed the extraction of the activities from the literature. First, this researcher designed keywords for the literature search and search space. These keywords were checked and commented by the other two authors. After that, the first author performed the search and enhanced the keywords if the results of that literature search indicated relevant other keywords. In total 3860 search results from 13

scientific literature databases were retrieved and scanned for relevance. Of course, a large number of these 3860 papers overlapped in the search results of these databases. The literature and extracted activities from the previous research (Janssens et al., 2010) were also included in the results.
The full text of literature that seemed relevant after a first check was scanned in detail for relevancy for this research. Finally, 42 papers were selected which include relevant information on ERP implementation activities i.e. the paper did describe and mention concrete ERP implementation activities.
From the results of this literature search it can be concluded that ERP activities are described in these types of papers:

1. Papers which used accepted ERP concepts from previous research for designing new theoretical models, for instance, critical success factors, stage theories and supplier implementation methods.
2. Papers which portrayed and analysed real implementation cases.
3. Papers which combined theoretical models with empirical data.
4. Papers which contained ERP implementation activities but where the origin of the list of activities was not always stated.
5. Papers which showed activities for implementation of enterprise systems.

From these 42 relevant papers, shown in Appendix 4.4, a list of 484 ERP implementation activities was retrieved.

4.5.5 Refining the collection of ERP implementation activities

In the extracted activities from scientific papers of course synonyms and homonyms occurred. For example, one paper would mention 'training of users' whereas other papers would mention 'user training'. Also, homonyms exist, for example 'redesign' might stand for 'business process redesign' or for 'infrastructure redesign'. To refine the total list of activities from synonyms and homonyms, five participants with appropriate knowledge in the ERP field were selected and received formal criteria, instructions and rules how to detect synonyms and homonyms.
The five participants consisted of two of the researchers and three Master of Science students which were preparing their thesis on an ERP implementation subject and also had practical business experience, as students from the Open Universiteit usually have. The detailed explanation of the used procedure is available through the authors.
The participants which should refine the list received the total list of 484 activities with the instruction to indicate which activities, should be deleted, because of being synonym with another activity or activities. Activities could have an identical or nearly identical name, for instance 'test reports' and 'reports testing'. Activities could be a summary of other activities, for instance 'plan project' and 'schedule activities', 'add resources', 'schedule resources' et cetera. Activities could be formulated too gen-

erally whereas another activity would be more specific, for instance 'define objective' versus 'define project objectives'. Finally, activities could be deleted by interpretation of the meaning of the activity. For instance, when an activity seemed out of scope. According to these instructions they detected the synonyms and homonyms independently from each other.

One of the researchers processed the results of the five participants according to in advance declared rules to form the final list. Application of these rules to the results of the five participants led to a condensed list of 232 activities. Finally, the same researcher checked in detail this list of 232 activities and removed 27 clearly inappropriate activities, which led to a cleaned list of 205 activities suitable for grouping. Also, these activities were renamed in a consistent formulation, i.e. verb + noun(s), for instance 'definition of scope' would become 'define scope'.

Table 4.6 shows the refined list of ERP implementation activities.

Table 4.6 Refined list of ERP implementation activities

1. Deliver conceptual detailed plan for business processes re-engineering	106. Add people to accommodate learning and shakedown needs
2. Deliver conceptual detailed plan for the new IT infrastructure	107. Identify the operational needs, business drivers, strategic plans and other factors that will define the scope and objective
3. Develop documentation	108. Identify expectations for benefits realisation, magnitude of change, change ownership, process redesign and functionality delivery options
4. Select ERP package	109. Evaluate the organisational baselines
5. Realise communication (plan)	110. Review functional and technical requirements
6. Put in place process management (business process redesign)	111. Perform cultural and workforce skill evaluations
7. Put in place project management	112. Make final changes to business processes, policies and procedures and system builds
8. Integrate system	113. Determine a post-implementation audit
9. Test system	114. Analyse and review current system
10. Carry out cultural and structural changes	115. Realise resource determination
11. Define project objectives	116. Put in place reporting mechanisms
12. Collect vendor consultant information	117. Repair system
13. Find out need analysis	118. Make investment decisions
14. Evaluate vendor and consultant alternatives	119. Perform cost-benefit analysis
15. Evaluate IT infrastructure	120. Choose appropriate technology
16. Perform feasibility study	121. Analyse suitability of innovation for the firm
17. Finalize contracts	122. Observe user resistance
18. Define project scope	123. Increase usage of systems
19. Establish implementation teams and timetables	124. Carry out systems modifications to fit user needs
20. Determine the most appropriate approach to implementation	125. Integrate functional units
21. Perform detailed gap analysis	126. Correct flaws
22. Identify complementary solutions	127. Realise organisational integration
23. Construct of prototype	128. Recognise territorial walls in the organisation
24. Convert data	129. Implement radical changes in the organisational structure, the reward systems and the organisational culture
25. Take care of clarity of work procedures	130. Prepare IT infrastructure
26. Train users	131. Implement a configurator (add-on module)
27. Make general design	132. Produce, review and sign-off on a business blueprint
28. Implement the solution	133. Produce and review design specifications for configuration
29. Start preparation	134. Configure business processes
30. Perform organisational readiness assessment	135. Perform initial review and test of configured business processes
31. Configure baseline system	
32. Fine tune system to meet all of the business process requirements	
33. Build the business case	
34. Set expectations	

35. Measure results
36. Celebrate success
37. Identify vision and targets
38. Start software design and development
39. Plan integration
40. Define system requirements, its goals and benefits
41. Perform an analysis of the impact of adoption at a business and organisational level
42. Select a consulting company
43. Analyse functionality, price, training and maintenance services
44. Analyze return on investment of selected product
45. Customise or parameterize and adapt ERP package
46. Create detailed project plan
47. Perform new process design mapping
48. Realise technical development
49. Bring ERP-module(s) live
50. Create and issue project charter
51. Review and refine implementation strategy
52. Establish project team working environment
53. Identify and plan the primary focus areas to be considered
54. Create a visual model of the business' future state
55. Analyse organisational processes and compare them with the procedures embedded in the ERP package
56. Specify functional and data requirements for modules that need reprogramming
57. Produce a technical specification of the functional and data requirements for modules that need reprogramming
58. Develop and verify software code for modules that need reprogramming
59. Test all modules against requirements as well as quality parameters
60. Set-up of the steering committee
61. Map the business processes onto the ERP functions
62. Make detailed design
63. Perform population of the test instance with real data
64. Build and test interfaces
65. Write test reports
66. Build networks
67. Install desktops
68. Make requirements review
69. Understand system
70. Address business issues
71. Map workflow
72. Define user acceptance criteria
73. Test integration
74. Test user acceptance
75. Train project team members and do acquisition of supportive skills
76. Execute change management plan
77. Customize software
78. Integrate software built-ons and/or legacy systems
79. Rollout and startup
80. Fix bugs
81. Rework
82. Add hardware capacity
83. Carry through process and procedure changes
84. Realise user acceptance, retrain and additional training
85. Carry through organisational changes to accommodate learning and shakedown needs
86. Format budget
136. Perform go-live check
137. Prepare production environment
138. Migrate data to production environment
139. Realise role-user assignment
140. Approve system and organisational readiness to go-live
141. Create go-live strategy
142. Perform management of expectations
143. Define the architecture
144. Dedicate resources
145. Educate on new business processes
146. Communicate interdepartmentally
147. Take care of interdepartmental cooperation
148. Develop systems design
149. Determine the level of previous reengineering
150. Create the vision
151. Create a feature/function list
152. Create a software candidate list
153. Create the request for proposal (RFP)
154. Have the finalists demonstrate their packages
155. Justify the investment
156. Negotiate the contract
157. Run a pre-implementation pilot
158. Review the pre-implementation process to date
159. Install and test any new hardware
160. Install the software and perform the computer room pilot
161. Ensure that all data bridges are sufficiently robust and the data are sufficiently accurate
162. Document policies and procedures
163. Bring the entire organisation on-line, either in a total cutover or in a phased approach
164. Implement knowledge management
165. Audit systems
166. Migrate software
167. Draw up a business strategy and model
168. Gain input from agency representatives through workshops
169. Implement the software in phases
170. Collect the relevant information about ERP systems
171. Identify the project characteristics
172. Distinguish the fundamental- and means-objectives
173. Screen out the unqualified ERP systems
174. Collect information and eliminate unqualified alternatives
175. Evaluate and aggregate external professional data to obtain objective ERP suitability
176. Evaluate and aggregate the data from interviews to obtain subjective ERP suitability
177. Select the ERP project with maximum final ranking value
178. Define the project sponsor and project manager
179. Obtain top management support
180. Develop functional specifications, project deliverables and metrics
181. Define areas of responsibility
182. Make a plan for project reviews
183. Finalise the detailed planning process
184. Finalise requirements definition stage
185. Send request for proposal (RFP) to potential vendors
186. Make change control procedures
187. Get tools to measure performance results
188. Perform custom programming (if required)
189. Check references of ERP vendors
190. Swap software previously used with the new ERP pack-

87. Identify modules needed
88. Select implementation partner
89. Identify data and system interfaces
90. Make inventory of existing hardware and software
91. Make test reports
92. Decide on whether to proceed with ERP or not
93. Staff up to handle temporary inefficiencies
94. Define very clear outcomes
95. Define performance metrics
96. Develop education and training strategies
97. Communicate ERP plan to the enterprise
98. Create super-users and troubleshooters
99. Inform suppliers and customers
100. Define key performance indicators and process of measurement
101. Perform current state analysis
102. Make initial plans for how the system will be rolled out, supported, and maintained, upgraded, etc.
103. Carry through organisational changes and/or incentives related to enterprise system and/or organisational improvement
104. Make decision to proceed, approve project plan
105. Realise problem resolution
191. Respond to enhancements that were not included in the original implementation
192. Prepare a final report
193. Provide an outline plan and costing
194. Analyse the enterprise model
195. Perform system deployment
196. Bridge the legacy system and clean up suspect data
197. Train senior management
198. Manage effective relationships and leading teams
199. Manufacture simulation exercises
200. Perform shop floor communication with line workers
201. Give staff training
202. Perform system investigation
203. Make demand analysis and set target
204. Prepare data
205. Run test and customise

4.5.6 Grouping of the collection of ERP implementation activities

Information about the dependency between the ERP implementation activities can be retrieved from three sources: literature, documents from ERP projects and persons who have sufficient knowledge and experience of ERP implementation projects.
As in the researched literature, except with regard to phases, no formal non-phased groupings were detected, only documents in ERP projects and persons with sufficient knowledge are in this case appropriate sources of information.
Documentation can be disclosed by detecting from literature collected activities in the actual documents from completed ERP implementation projects and analysing the relationships between these activities, for instance by the viewpoint of subprojects. The advantage of the document method is objectivity and reproduction of the results. However, disadvantages are: a large number of projects needed, the necessity of constructing and verifying a proper method for analysing the project documentation, the in general for research purposes difficult access to these projects and the large cost and duration needed to perform the research itself.
In fields where knowledge in the decision-making processes is rare and incomplete, expert consultation is often used (Jacobs, Moll, & Kusters AC (Aarnout); Trienekens, 2007). However, it can be expected that no single expert exists with every necessary knowledge needed to form the collections of the retrieved ERP implementation activities. Even if this expert would exist, the only way to detect this expert is by testing the expert against the knowledge which is yet to be retrieved, which of course is a paradox. On the other hand, a group of experts will have overlapping knowledge (Kasvi, Vartiainen, Pulkkis, & Nieminen, 2000) and therefore can provide the necessary input if an appropriate consultation method is used and the quantity of the group is sufficient. Gustafsson and Ollila (2003) showed the characteristics of the consultation

methods for groups of experts in light of the communication media theory. These consultation methods are: questionnaire, interview, workshop and Delphi. They also designed an application typology for these consultation methods. In the case of topics which relate to multiple disciplines and are ambiguous, they recommend as consultation methods Delphi or workshops. Based on their typology, workshops or Delphi would be proper consultation methods in this research, as the implementation of ERP projects is a topic which relates to multiple disciplines and also is still an ambiguous topic. They also conclude that if the topic is uncertain, Delphi would be the proper method.

In contrast to analysing project documents, in our research use of a group of experts has the advantages of easier access to this information source, of a smaller duration of the research itself and lower costs. Although it can be expected that the results will be influenced by the composition of the group of experts, and therefore the results will be less objective than analysis of project documents. Nevertheless, by proper selection of experts and use of an adequate workshop or Delphi consultation method, the quality of the results will be sufficient to form a first taxonomy. Therefore in this research experts were invited to group the ERP implementation activities into meaningful collections which form the taxonomy. A second reason for choosing experts to form the collections is that we tested in our previous study (Janssens et al., 2010) a method for detecting activities that normally occur in an ERP implementation project and a method for grouping these activities into collections of ERP project activities. This study shows that both the selection of activities as well as the used method of grouping is appropriate and relevant results can be obtained.

To be able to group coherent collections of these activities, it was necessary to select an approved consultation method by which experts could model the groups of activities. As suggested by Gustafsson and Ollila (2003) a Delphi method would be appropriate. From the outcomes of the previous research (Janssens et al., 2010) it was concluded that the metaplan technique, which is a form of Delphi, is a suitable technique for grouping ERP activities, but unfortunately also has some practical limitations. The number of 205 activities in this present research and the fact that experts are hard to persuade to participate in a group session like metaplan, indicated that the metaplan technique would cause practical limitations. The grouping of these activities is dependent on time and place, and it can be expected that the metaplan session would be very hard to organise and have an unacceptable long duration for the participating experts. The researchers expected that if experts could perform the grouping whenever they want and wherever they want, the willingness to participate would be higher. As shown by Howard as well, support of this process by a Group Decision Support System (GDSS), which can support grouping in different locations and/or at different times, leads to the same quality of results (Howard, 1994). Also in our research, the Delphi aspect should be integrated into the GDDS.

The number of 205 established activities implies the need for a formal technique. For this type of grouping a card sorting technique, which will be described in detail fur-

ther on, is appropriate, as card sorting is a simple method for establishing a taxonomy which cannot be inferred from objective sources of information. Card sorting has proven its usefulness in many concept mapping studies (Trochim, 1989). If one human individual does a card sorting, bias and limited knowledge will influence the result. Judgment by several individuals and group interaction will improve the quality of the results. Unfortunately members of freely interactive groups are often dissatisfied with group interaction (Howard, 1994). According to Howard, a Nominal Group Technique (NGT) improves the output and satisfaction of the group members (Howard, 1994). Therefore in our previous research (Janssens et al., 2010), the metaplan technique for the grouping was chosen. The metaplan technique uses card sorting and can be considered a Nominal Group Technique (NGT). However, card sorting as a regular technique does not contain the Delphi aspect. Fortunately, Paul (2008) combined the Delphi method with the card sorting method. She showed that the combination of the Delphi method with the card sorting method results in better grouping quality when compared to regular card sorting. Paul also showed that the experts needed less effort, which in this research is very relevant as the number of activities to be grouped is large in comparison to regular card sorting. Therefore this research adopted Paul's Delphi card sorting method and used Websort as a supporting GDDS.
In the next sections, the selection of the experts, the concepts of regular card sorting and the differences with and advantages of the Delphi card sorting technique and the selected tool for the card sorting are discussed.

Experts
To perform the Delphi card sorting about 8-10 experts as participants are necessary (Paul, 2008). The experts in the present research had to meet the following requirements:

- Minimal five years of experience as a manager in ERP implementation projects.
- Knowledge and experience of the complete ERP implementation issues and not only on a special issue.
- Sufficient English knowledge to understand the descriptions of the activities from the activity collection and to be able to name categories
- A professional reflection level indicated by at least a completed Bachelor degree.
- Experience in ERP implementations in the Netherlands, because of the risk that Paul recognised, that too heterogeneous a group could lead to an unstable model (Paul, 2008).
- Knowledge of and experience in implementation of large ERP applications like SAP, Oracle, BAAN of Peoplesoft, whereby several modules were implemented.

- Originating from different organisations.

One of the authors approached one expert individually by within their mutual organisation; other experts have been selected and approached using LinkedIn (http://www.linkedin.com). LinkedIn provided sufficient access to appropriate experts.
Ultimately, 11 experts that met the requirements agreed to participate in the Delphi card sorting study (Paul, 2008). This number satisfies the criterion from Paul (2008) that 8-10 experts are needed to perform a Delphi card sorting study.
This group of 11 experts consisted of six managers and five consultants. In this group, three experts worked for Dutch ministries and eight for business organisations. Of the 11 experts, six had five to 10 years of the required experience in ERP implementation projects and five had 10 or more years of experience. All experts worked in the Netherlands.

Regular card sorting versus Delphi card sorting
Card sorting is a simple method for establishing a taxonomy which cannot be inferred from objective sources of information. In card sorting several individuals with relevant knowledge of, or attitude to a subject, classify items into categories. Two types of regular card sorting exist:

1. open card sorting
2. closed card sorting

Open card sorting is mainly used to obtain a classification of concepts (taxonomy) when uncertainty exists concerning what that classification might be. At open card sorting, each participant receives a full deck of unsorted cards and must form relevant piles and also name these piles at his or hers discretion. The results of all participants are statistically evaluated, and from the degree in which the cards appear in the same collection with other cards, the relevant categories are determined. One or more individuals must determine by good judgment the final names of the categories to find the best match.
The closed card sorting method is used to validate concepts and to add information to existing taxonomies. In closed card sorting categories already exist. Each participant receives a full deck of unsorted cards and must sort these cards into the existing categories. To form the definite classification of the cards into the existing categories of all participants, also statistical analysis is used.
In this study into the grouping of activities for ERP implementations, only open card sorting can be a useful method, as there are no relevant groupings of ERP implementation activities in scientific research

Unfortunately also some disadvantages exist for the use of regular card sorting within our research:

- Generally, every participant starts from the same situation, i.e. one uncategorized stack of cards. As a result, every participant has to sort all the cards. Every participant has tot sort the obvious ones as well as the difficult ones. As a consequence, the time needed for sorting a lot of cards will be quite long. Unfortunately for retrieval of this ERP implementation knowledge, highly paid experts are needed, who will not easily volunteer to participate if the effort required is too demanding.
- The open card sorting method has the disadvantage that the final naming of the categories is rather subjective and will not be checked by the participants themselves.
- It is recommended for both the closed card sorting and the open card sorting that approximately 20-30 persons should take part in the sorting to obtain a 0.90-0.95 correlation in the results (Tullis & Wood, 2004). It is difficult to arrange such a group of experts.
- Because every participant starts with an uncategorized stack of cards, the sorting is an activity which needs full concentration and takes up much time. Considering the number of 205 activities in this research, the use of a standard open card sorting will take considerable time and concentration, which could influence the quality of the final result and as already mentioned, the eagerness of the participants to cooperate. For standard card sorting, it is recommended to sort a maximum of 50-100 cards. Considering that in this research 205 cards (=activities) must be sorted, this undoubtedly will reduce the eagerness of the participants to participate and lead to practical concentration problems.

Nevertheless, standard card sorting as a method has also some important advantages:

- It is a simple method easily understood by participants and can easily be applied.
- It is a low-cost method to achieve a classification of concepts (taxonomy).
- Participants can do the sorting independently of each other in time and place.
- The method can be applied rapidly. Especially if physical cards are used, the preparation is very easy and inexpensive.

Card sorting as a method should be understood as a formal method to obtain information about specific opinions of people. In this regard, the card sorting method holds the same disadvantages as conducting surveys. The results of surveys, as well as card sorting methods only, provide a common view of participants about the research subjects. That view is always constructed by past experiences from the partici-

pants from the study. An objective assessment of the outcome of such methods is not always possible (Robertson, 2001). It is important to value the results keeping in mind that they represent the opinions of a selected group of people. Nevertheless, card sorting is a useful method for retrieving a new taxonomy.
After research on a variation on the open card sorting method, Paul (2008) proposed a new open card sorting method. She named this variation 'modified Delphi card sorting method'. This method deviates from the closed card sorting method in that the participants, except for the first participant, do not start with a pile of unsorted cards, but receive already sorted and named piles from their predecessor. The first participant performs the initial sorting of cards and provides each category with its initial name. The next participant and subsequent participants will continue improving this initial sorting and categorization of their predecessor as they please. Which means that they can move cards from one pile to another, start new piles, delete piles and make changes to the naming of piles. The idea behind this adapted method of card sorting is that towards the end, fewer changes will be applied by the participants and they only need to reflect on the difficult issues. The model towards the end will stabilise. The final result of the sorting will be the result of the last participant. Therefore, there is in principle no further statistical analysis necessary to find the appropriate groups of cards. Furthermore, it is not necessary to 'subjectively' decide on what the final name of each category should be, as the participants have changed the names if they thought them inappropriate.
Naturally, this new method also bears some disadvantages:

- The lead time of the sorting will be longer compared to regular card sorting because participants cannot sort simultaneously and are dependent on each other's results.
- Undoubtedly the sequence in which the participants participate in the sorting procedure can influence the outcome.
- If the last or one of the last participants has a complete deviating opinion on how the cards should be sorted, then the result of that final sorting might not be optimal or even useless. However, the results from the previous rounds still can be of use.

In this research, we tried to cope with these last two disadvantages by analysing the results from every participant and comparing them to each other.
In her research Paul (2008) also concluded that the participants of the Modified-Delphi card sorting method were more relaxed during sorting and had more attention in the communication with, in her case, the examiner than the participants of the regular card sorting. Apparently, the concentration impact of the modified Delphi card sorting method (that is to say if the participant was not the first) is lower than the standard card sorting method. The participants in the Modified-Delphi card sorting method therefore can put more effort into the sorting itself. The participants only

have to focus their attention on the items that in their opinion are not in the correct pile and/or only need to consider new categories that do not yet exist. Although participants don't know what their participating rank is, it is expected that the model towards the end of the cycle will become more stable and of higher conceptual quality. Theoretically, each participant will improve the model of its predecessor. It can also be expected that the shift of cards at the beginning of the sorting cycle will be more numerous than at the end. Towards the end, participants will mainly concentrate on items that are difficult to place in a category; for reasons of ambiguity or because there are several categories in which an item logically could belong. This implies that allocation of an item in a category in a later round is of higher importance than allocation in the first rounds, which also applies to the creation and naming of categories.
In her research Paul (2008) showed that the quality of the final model by Delphi card sorting was better than the quality of a model through the regular card sorting method.

Tool
In the field of ERP implementations, it is difficult to encourage experts to participate in scientific research. Therefore important requirements for the method and supporting tools are the minimization of time and effort to be spent by experts and also independence of place to perform the sorting. Taking these requirements into account, an internet based card sorting tool is a suitable solution. Hence Websort (http://www.websort.net) has been selected, which supports card sorting and specifically Delphi card sorting. The tool is easily accessible, and the functionality is user-friendly. The experts reported no problems using it, as they also were invited to make remarks in Websort. Also, the results of the sorting (per expert) could be easily exported and further processed in a spreadsheet.

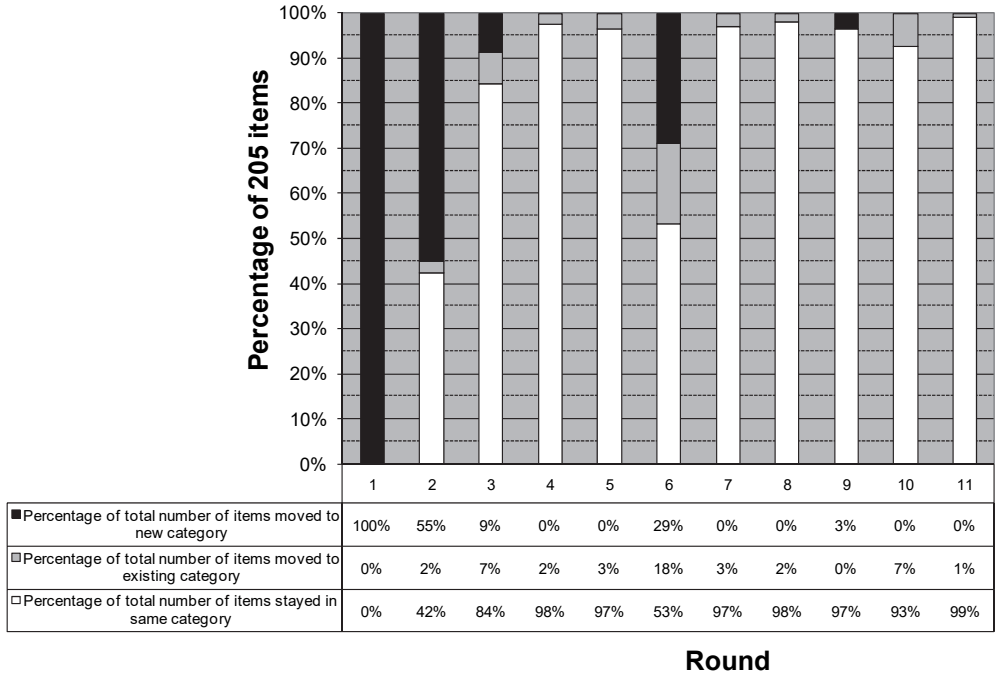
4.5.7 Grouping

The 11 independent experts used Websort to group the 205 ERP activities. The order in which the experts have executed the grouping, except for the first one, was random. The first expert was individually approached by one of the authors to be able to motivate and instruct this expert, as this expert had to perform the initial sorting of the 205 activities which is, of course, a time-consuming and complex task.
The first expert received the complete set of unsorted activities and was asked to group this set into relevant groups (piles) and label those groups with an appropriate name. The second expert received the anonymous result of the first expert and was invited to improve the grouping regarding relocation of activities between groups and changing group names and/or creating new groups. The 3rd to the 11th expert received the anonymous outcome of their predecessor and the same instructions as the 2nd expert. Websort provided sufficient information to be able to evaluate what

the experts changed between consecutive rounds. Figure 4.2 and 4.3 show a graphical representation of the changes between rounds.
Of course, the 11 experts drawn from the population of experts on this subject, influenced the outcome. If 11 other experts were selected the outcome would probably have been differed in some details. However, this is always the case in forming a taxonomy for the first time.
Besides that, although the order in which the experts participated in the Delphi card sorting was random, the order will have influenced the outcome. In case the 6th expert would have been selected last and also would have made many changes, round 11 could have shown a big deviation from the previous ten rounds, and it would not be possible to draw the conclusion that this expert had improved the model.
Every expert, except the first one, was influenced by his predecessor although unaware who sorted before him and the round in which he was participating. On the one hand, this is the intention of Delphi card sorting and improves the quality of the model, on the other hand, the influence of the predecessor narrows down the idea for a solution for the successor expert.

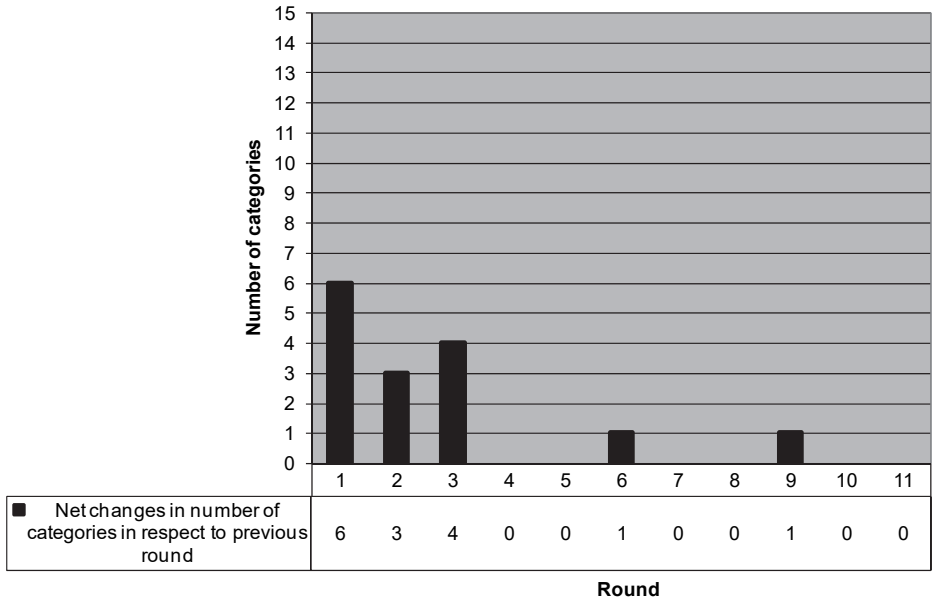
4.5.8 Results

Apart from the first expert, none of the experts knew in which round he was. Therefore it is remarkable that the results clearly indicate a quick stabilisation of the sorting between rounds 1 through 5 and even quicker between rounds 6 to 11, as shown in figure 4.2. Figure 4.3 shows the same stabilisation for the number of categories. The stabilisation seems to confirm the claim of the Delphi card sorting method that each round will improve the model. Although the experts did not know in what round they resided, each following expert needed to improve less on the results of his predecessor.



	1	2	3	4	5	6	7	8	9	10	11
Percentage of total number of items moved to new category	100%	55%	9%	0%	0%	29%	0%	0%	3%	0%	0%
Percentage of total number of items moved to existing category	0%	2%	7%	2%	3%	18%	3%	2%	0%	7%	1%
Percentage of total number of items stayed in same category	0%	42%	84%	98%	97%	53%	97%	98%	97%	93%	99%

Figure 4.2 Movements of items in categories per round



	1	2	3	4	5	6	7	8	9	10	11
Net changes in number of categories in respect to previous round	6	3	4	0	0	1	0	0	1	0	0

Figure 4.3 Net changes in number of categories in respect to previous round

Figure 4.2 shows a quick stabilisation of the sorting in rounds 1 through 5. The relocation of activities between the groups and the changes to the number and the naming of the groups decreases. However, the expert in round 6 made a considerable change to the model. Round 7 to 11 again show increased stability. The experts 7 to 11 apparently accepted the significant changes made by expert 6 and only made improvements upon these changes. The graph in figure 4.2 might lead to the conclusion that the sixth expert has largely messed up the model of his predecessors, but detailed analysis of his changes shows that the 6th expert has refined some groups. This expert kept nine groups the same, split two and combined two into one. Apparently, this expert changed the level of abstraction. This change of abstraction also can explain why round 7 till 11 again show increased stability. The experts in these rounds accepted this more detailed level of abstraction from the sixth expert and improved this level.
Given the fact that, except for the first expert, the experts themselves did not know in what round they resided and all started from a given grouping, it must be concluded from round 5 and 11 where stability in the model clearly occurred, that no possible optimal classification/grouping of activities exists. The groups are determined depending on the point of view of an expert, i.e. the level of abstraction in which he performs the grouping. If other experts agree with this level, this leads to an acceptable; that is to say, stable grouping.

4.5.9 Comparing results with a previous grouping

In the exploratory research (Janssens et al., 2010) also activities within an ERP implementation project were extracted from scientific papers and grouped using the metaplan method. The activities and groups retrieved in that study do not fully correspond to the activities and groups in this study. However, at first sight, there seemed so much similarity between the two studies, that comparing the grouping from both studies should provide useful insights. Therefore the activities from the first research (Janssens et al., 2010) were matched with the activities in this research. Two authors conducted this matching independently of each other and checked their matching with each other afterwards. Next, it was determined to what extent activities ended up in the same collection of activities, regardless of the name of the group.

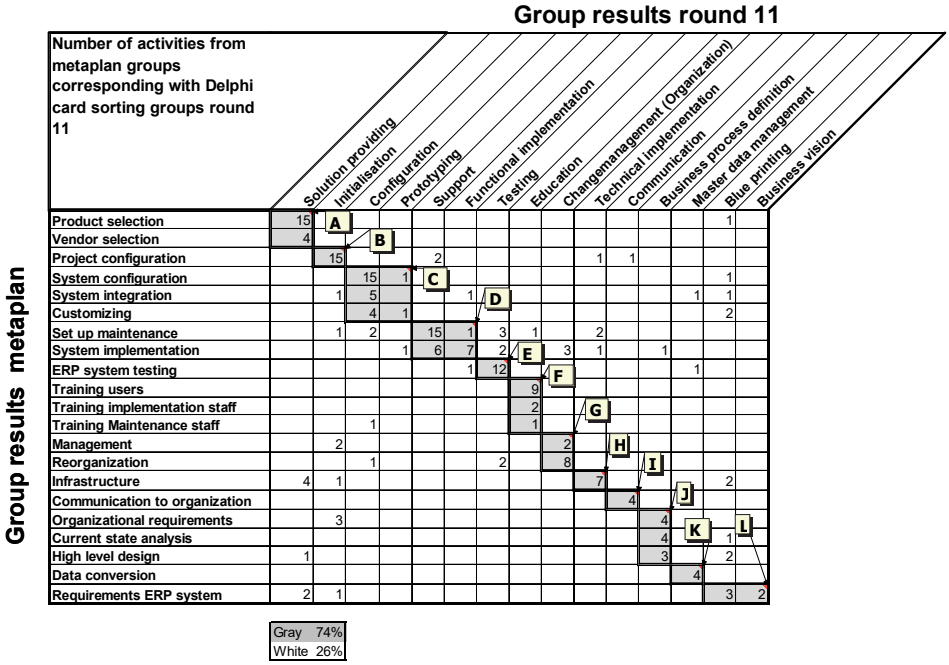


Figure 4.4 Comparison sorting round 11 with metaplan sorting

Figure 4.4 shows the results of the comparison of the groups with the exploratory research (Janssens et al., 2010) and the outcome of the 11th round. The results from the 5th round were not used, because the model of the 11th round is a more detailed model than the 5th round, and not significantly different from the 5th round. Figure 4.4 shows on the y-axis the groups from the exploratory research. The x-axis shows the groups from this. The cells in the matrix represent the number of the 192 activities which were assigned to a group from the exploratory research and to a group from this research. For example from the 17 activities from the metaplan group "System configuration", 15 activities were classified in the "Configuration" group in round 11, one in the "Technical implementation" group and one in the "Blue printing" group. To further explore the overall similarity of the grouping between the two sorts, the matrix has been sorted in such a way that the cells with the largest numbers of matching activities were moved to the diagonal sorted in decreasing order. I.e. decreasing from the top left to the bottom right in the matrix. After that, borders were drawn around adjacent groups of cells on the diagonal which contained the most activities to form groups of groups between the two grouping's which are closely related to each other, or in other words, which are very similar. As shown in the matrix 12 groups were formed containing 74% of all activities, which indicates that there is a major similarity between the grouping of the exploratory research and this research. Table 4.7

shows a list of these similar groups of groups with a proposed name by the authors, taking into account the nomenclature in the two studies.

Table 4.7 Proposed taxonomy of ERP implementation activities

A	Software and vendor selection
B	Project initialization
C	System configuration
D	Implementation and support
E	Testing
F	Training
G	Change management
H	Technical implementation
I	Project communication
J	Business Process Design
K	Data conversion
L	Blueprinting

4.5.10 Conclusions Study 3

Experts agree upon coherent groups of activities that occur in ERP implementation projects. However, the groups are dependent on the level of abstraction of the view the experts use when grouping activities. Though very similar, the results of our previous exploratory research and the results of this research from round 5 and 11 provided different groups of activities. Analysis of the data shows that participant number 6 changed the level of abstraction into a more detailed one. Also, the comparison of the grouping from our previous research with the results of round 11 show that there is a great similarity, dependent on the level of abstraction. Therefore we assume that the adopted level of abstraction by a participating expert is an important factor. We assume that there is no ultimate correct grouping of ERP implementation activities. Nevertheless, the combined high-level grouping from the exploratory research and this research is a first by experts verified grouping of activities in highly related activities within an ERP implementation project independent of phases.
As a result of the expert grouping and the comparison with the previous research, table 4.7 can be considered as a first proposal for an expert based taxonomy of ERP implementation activities at a fairly high level of abstraction. This taxonomy can serve as a base for further research into ERP implementation projects and can support the management of these projects.
The rapid stabilisation in round 1 to 5 and 6 to 11 seems to confirm the claim of the Modified Delphi card sorting method that each round will improve the model. Although the experts did not know in what round they resided, each following expert needed to improve less on the results of his predecessor. As shown in figure 4.2 in round 5 and 11 groupings exist with stable cores as only a few percent of the activities

still move during round 3 to 5 and even less during round 6 to 11. These few percent can be regarded as the 'noise' around the stable cores of activities in the groupings. Although some experts commented that they missed a particular detailed activity in the set of 205 activities, they did not indicate that they, therefore, were unable to form a needed group. Apparently, all expected groups could be formed using the available 205 ERP activities.

The results, participation and comments from the experts show that Delphi card sorting appears to be a practical method for retrieving this type of information and Websort is a suitable supportive tool. The willingness of the invited experts to participate in this online Delphi card sorting was high. All experts who were willing to participate also finished the sorting. This willing participation might confirm the assumption that an appropriate method and tool would stimulate the participation of experts and the actual sorting. The possibility for the experts to sort whenever and wherever they wanted, and the user friendliness of the tool might be important factors. Websort also provided functionality to an expert to comment on his sorting and the tool itself. In these comments, none of the experts complained about the method or used tool for the sorting. Paul (2008) also observed that performing Delphi card sorting required less effort from the experts than regular card sorting. In this research, we had no opportunity to validate this observation, but it might have been a factor which influenced the willingness of the experts to finish the sorting.

The resulting taxonomy in this research is a taxonomy solely based on expert judgment. This taxonomy should, therefore, be confirmed and enhanced by the use of empirical data from ERP implementation projects.

Also, more groups of experts should perform the grouping to confirm the fact that there are various levels of abstraction possible.

Further research into this taxonomy should confirm and define the various levels of abstraction, as this and our previous research show different levels. Also, it would be appropriate to research into the reasons or circumstances which lead to the adoption of a used level of abstraction.

The results from this research can also be used in other ERP research areas for instance to confine research into well-defined topics within an ERP implementation project.

4.6 Conclusions from Study 1, 2 and 3

In this section, we will recapitulate conclusions from Study 1, 2 and 3.

In Study 1 we searched for factors which might influence the complexity of the integration process of ERP systems and non-ERP systems. We retrieved these factors from literature and confirmation by experts. The by experts confirmed top five rated factors in this study are:

1. Number of organisations that need to be integrated
2. Number of applications.
3. Number of project owners and stakeholder groups
4. Possibility to develop custom adapters
5. Willingness of employees to share control & ownership of processes

The research approach was a typical deterministic research complexity approach which intended to discover key variables by using historical data from literature and opinions based on past experiences of experts.
The result is a useful list of factors which practice and research can use as an instrument for recognition and structured discussion about the important factors which influence the complexity of integration. However, uncertainty remains whether this list is complete and whether the importance of each factor is correct. Also, relations and therefore correlations between factors should further be investigated.

By Study 2 and 3, we intended to gain better insight into the complexity of ERP implementations by setting a basis for better determining the size of an ERP implementation. The premise is that by being aware of standard clusters of activities for ERP implementations, estimation of costs and time could be better supported when starting an ERP implementation. Therefore both studies determined which activities usually are performed within an ERP implementation project. These studies started with the extraction of activities from literature. In Study 2 the clustering of these activities into meaningful clusters was performed by a group of researchers. In Study 3 this clustering was performed by experts in the field. Study 3 complements Study 2. The resulting clusters retrieved by Study 3 are:

1. Software and vendor selection
2. Project initialization
3. System configuration
4. Implementation and support
5. Testing
6. Training
7. Change management

8. Technical implementation
9. Project communication
10. Business Process Design
11. Data conversion
12. Blueprinting

Study 2 and 3 also can be regarded as studies performed by a deterministic research complexity approach. In this case, the key variables are the named activity clusters. Study 2 and 3 also used historical data from literature and past experiences by experts. The conclusions from Study 3 clearly state that the level of abstraction is also an important factor which complicates the design of the correct division of activities in clusters. Also, more research is needed to confirm these clusters further.

By Study 1, 2 and 3, we tried to extract variables from past experiences to provide future ERP implementations in practice and research into ERP implementations with guidelines and variables. Through the conclusions of all three studies, it became clear that the results need further confirmation and deepening by analysing more historical data and confirmation by experts. In all three studies, the extent cannot be indicated to which further confirmation and deepening will be sufficient.

4.7 Added value of a deterministic research approach

In the previous sections, we discussed research goals, methodology, results and conclusions of Study 1, 2 and 3 separately. However, in this section, we will discuss abilities and limitations of these (implicitly) deterministic complexity research approach based studies. I.e., what have we learned from these deterministic complexity approached studies about a deterministic complexity research approach for ERP implementations?

As a structure for our discussion we will use the abilities and limitations of a deterministic complexity research approach that were previously discussed in Chapter 3:

Abilities

- This research approach prevails in current ERP research and has retrieved useful results.
- Methods and research areas for this type of complexity research approach into ERP implementations are well-known.
- A vast amount of this type of research is available as a base for future research.

Limitations

- It is difficult to model the behaviour of social phenomena like ERP implementations.
- It is difficult to take into account unexpected changes in the ERP implementation field, by which the research results may become obsolete.
- It can be difficult to retrieve historical data about ERP implementations required for keeping your insight up-to-date.

Ability: This research approach prevails in current ERP research and has retrieved useful results
As is discussed in our three studies, we consider the results and conclusions of all studies useful. The retrieved guidelines from these studies can be used for practice as well as for research as support for managing and exploring ERP implementations.

Ability: Methods and research areas for this type of complexity research approach into ERP implementations are well-known
The research methodologies we thought most appropriate for these three studies are comparable to qualitative research methodologies in mainstream research into ERP implementations. Mainstream research which we consider as mainly based on a deterministic complexity research approach. For our three research studies, we did not need to design new or adapt existing research methods.

Ability: A vast amount of this type of research is available as base for future research
For the research in our three studies, we could easily base ourselves on existing research of deterministic complexity approach type. In Study 1 we retrieved a list of complexity factors from existing literature and similar in studies 2 and 3 we retrieved a list of ERP implementation activities from existing deterministic type based research. Retrieval of sufficient papers for the necessary information was not a problem.

Limitation: It is difficult to model the behaviour of social phenomena like ERP implementations.
In all three studies, we tried to find guidelines for research and practice. In Study 1 we identified factors which might influence the complexity of the integration process of ERP and non-ERP systems. We tried to distill factors from past experiences. However, these factors do not take into account situation-dependent behaviour from people, departments and organisations. In our research, these are considered black boxes who hopefully will react similarly in similar situations. The factors we defined should be sufficient to predict the complexity of a specific situation. The social interaction which influences the complexity of an integration process in a specific situation is not accounted for.
In Studies 2 and 3 we tried to model the structure of the activities which have to be performed in an ERP implementation. Also, in this case, we did not take into account the influence of social interaction when these activities are executed. After all, activities are performed by humans. Therefore in our design for Studies 1, 2 and 3, which implicitly was based on a deterministic complexity research approach, we ignored social factors to find 'deterministic' models which could be used for prediction and design of ERP implementations.

Limitation: It is difficult to take into account unexpected changes in the ERP implementation field, by which the research results may become obsolete.
Also, as discussed in Chapter 3 regarding deterministic complexity approaches, methods derived from the results of a deterministic complexity research approach are always based on observations from the past. In our three studies, we asked experts about their observations from the past and used literature to retrieve our factors. Therefore if the results from these three studies will be used in practice for decision-making, these decisions, supported by deterministic based ERP complexity handling methods, will be based upon past events from other implementations. As a consequence, if a decision has to be made upon an event which never has occurred during the past and therefore has not been incorporated in the model derived from this past, there will be a possibility that the solution suggested by the method will be wrong or that no solution exists. A limitation of the deterministic complexity research approach is that results and derived solutions for handling the complexity of ERP implementations will always be based on the past and not the present. Also, they will

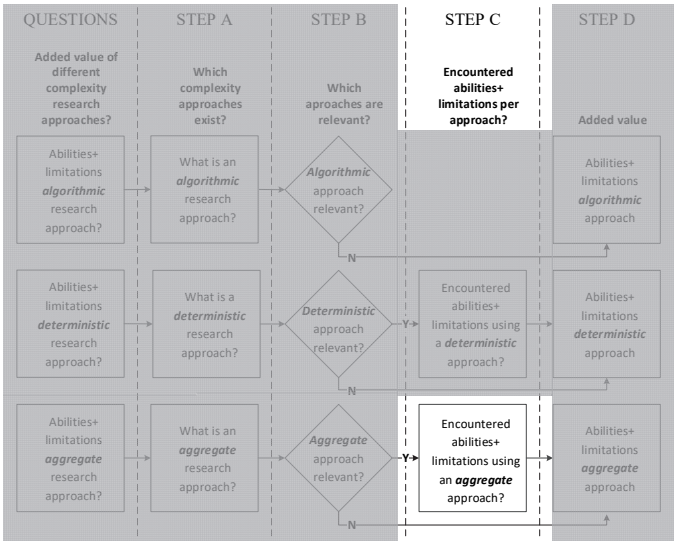
be based on experience from other organisations or situations. Therefore if there is a substantial gap between the past and present and/or a substiantal differenc between organisations or situations, the risk exists that the derived methods for handling ERP implementations are obsolete. However, this will only be revealed if the method fails.

Limitation: It can be difficult to retrieve historical data about ERP implementations required for keeping your insight up-to-date.
As suggested by the conclusions of all 3 studies, to enhance the reliability of the retrieved factors and clusters of activities, more data would be needed. As we expect to have discovered the most relevant papers which describe ERP activities, this means that when we need more papers, we have to wait until new papers come available. Also, it always will be difficult to find suitable experts who are willing to cooperate. And last but not least, if the quality of the results from our three studies should remain high, these studies should be repeated at regular intervals, for instance, every year based on new papers and experts.

Overall conclusion
As is discussed in our three studies, the results and conclusions of all studies are considered useful for practice as well as for research. Therefore we conclude that using a deterministic complexity research approach can be a useful approach and deliver benefits for practice and research. This conclusion is not surprising. As discussed before in Chapter 3, mainstream research uses this research approach. However, as far as we could ascertain from our retrieved literature, mainstream ERP research uses this approach implicit. Also, we confirmed by our studies the theoretical limitations of this approach for ERP implementation research.

CHAPTER 5: EXPLORATORY RESEARCH INTO THE AGGREGATE COMPLEXITY RESEARCH APPROACH FOR ERP IMPLEMENTATIONS

5.1 Introduction and research goal



In the previous chapter, we discussed the results of our studies 1, 2 and 3 which separately enhanced understanding of ERP implementation complexity. On this basis, we drew conclusions about the abilities and limitations of the deterministic complexity research approach (Manson, 2001). As discussed in Chapter 3, we presume that in general, ERP research is implicitly performed by a deterministic complexity research approach. However, we also concluded that an aggregate complexity approach seems a valuable additional approach. Nevertheless ERP research does not seem to apply this approach explicitly. Therefore, we performed an empirical study to obtain a first impression of the abilities and limitations of performing research by following an aggregate complexity research approach. We executed a case study through which we explored whether ERP implementations can show unexpected behaviour. Unexpected behaviour which is out of scope and out of control of the ERP implementation project itself.

In general, ERP implementations are designed and managed in practice as large projects. Project management assumes a substantially plannable and controllable route to its project goal. In our research, we consider that ERP project goal to be a technically as well as organizationally well-implemented ERP system. By applying solid risk management and proper planning and control, organisations expect that ERP projects will be able to reach this goal within the predefined borders (scope, quality, time and budget). Risk management, estimation tools, models derived from previous ERP implementations, knowledge of experienced experts, all contribute to this planning and management of future ERP implementations. The rationale for this is that the better these on experience based models become, the better the ERP implementation will take place as expected. Also, the less surprises will occur for the implementing organisation. The deterministic paradigm assumes that enhancing models based on previous experiences, in the end, will make them a better match for reality. Therefore if the quality of these deterministic models improves then managing an ERP implementation will improve.
However, if ERP implementations are considered from an aggregate complexity view (from Manson's typology (Manson, 2001) and Edmonds definition (Edmonds, 1999)), then it can be expected that an ERP implementation, being a social system, will despite proper planning and management show unexpected behaviour. Therefore it can be expected, that even in well managed and planned ERP projects supported by the best deterministic models possible, unexpected issues (behaviour) can arise. Unexpected issues which can't be solved within the current plans, scope and authority of the project.
In our literature search in Chapter 3 into complexity as a construct in ERP research, we discovered no research which was implicitly or explicitly performed with such an aggregate complexity approach in mind. Also, we discovered no research which provides guidance or insights on how to manage these unexpected issues. These authors performed no research by an aggregate complexity approach. Nevertheless, they discussed auto-organization, emergence and evolution, which are characteristics of an aggregate complexity approach.
As we concluded in Chapter 3 that the deterministic complexity approach dominates current research into the complexity of ERP implementations, we expect it to be worthwhile to investigate ERP implementations with an aggregate complexity research approach in mind.

The main characteristic of an aggregate complexity approach is its assumption that a complex system will show unexpected behaviour. Therefore in this part of our research, we will try to detect whether this unexpected behaviour of ERP implementations can be demonstrated in practice. As a result of this aggregate paradigm on the complexity of ERP implementations, we expect to find unexpected behaviour. Therefore, we try to verify in Study 4 that ERP implementations will exhibit such unexpected behaviour. To detect this unexpected behaviour, we performed exploratory

research into an actual project. There, we looked for issues which occurred, despite correct project preparation, planning and project management. Project management of the classic type was used, that is to say: planned and managed with the best deterministic tools and knowledge. We considered unexpected behaviour as unexpected issues. Issues which cannot be solved within the boundaries, abilities and authorities of an ERP project itself. From a deterministic complexity paradigm, all issues should be able to be solved within the scope and boundaries of the project with proper planning and use of relevant experience from other ERP implementations. In general, we searched for issues which needed a higher level of involvement and decision making from outside the projects sphere of influence and abilities.

Our goal in this study is to discover the existence of unexpected behaviour by demonstrating unexpected issues that are clearly out-of-scope of the implementation project and can only be solved outside of the project itself.
The second goal of this study is to get a first indication of the usefulness of the aggregate complexity research approach for ERP implementation research and practice.

We are looking for unexpected issues. So, first in the next section, it is necessary to define what we consider issues, what we consider unexpected issues and which types of unexpected issues we are trying to detect in our research.
Next, we will discuss our research design by discussing our research strategy, the case selection, validity and reliability.
Finally, we will discuss our results and conclusions for this case study and discuss abilities and limitations of applying an aggregate complexity research approach separately.

5.2 Issues and events

5.2.1 Introduction

By Study 4 we try to find evidence that unexpected issues can be detected if we examine ERP implementation projects by an aggregate complexity research approach.
To detect unexpected issues, we have to determine how to detect issues in the first place. Therefore in this section, we will discuss and define our conceptual model which explains how we consider an ERP implementation project handles issues. Also, we will discuss what types of issues we consider unexpected issues. The key concept in our model is the occurrence of events that issues cause in an ERP project. We expect by detecting events which occurred during ERP implementation projects; we should be able to detect the underlying issue or issues which caused these events. After identifying the underlying issues, we need to decide whether these issues are unexpected or not. In our research, we consider issues unexpected if the implementation project could not handle and solve these issues with the authority given to the project. Because the implementing organisation was not aware that these issues could arise at the time the ERP implementation project was set up. These issues need decision authority outside the project boundaries. Therefore when these issues occur, the formal decision authority for these issues will be unclear: explicitly or implicitly. In that case, **improper decisions** can be taken, **unacceptable decisions** for stakeholders can be taken, or **no decisions** at all can be taken. These decisions all can result in unsolved issues; the situation is not in control. We expect that these unsolved issues are most likely to generate new events and therefore these issues keep on recurring in the project. Hence detecting recurring issues in an ERP implementation might lead us to detect unexpected behaviour.
We will try to determine for these recurring issues whether the authority for solutions was clear and whether the authority for solutions was inside or outside a project. By retrieving information about an issue itself (whether an issue was solved or not and by which authority it was solved and/or should have been solved), we will determine whether an issue can be considered an unexpected issue.

In the next section, first, we will discuss a conceptual model by which we consider the handling of issues in an ERP implementation project. Next, we will discuss which types of issues we wanted to retrieve in our case.

5.2.2 Issue and event handling model

During the progress of an ERP implementation project, it is to be expected that issues will arise. For example, an issue could be the refusal of a department to cooperate in an ERP implementation project. This refusal causes the project to fail.
An issue can be revealed by one or more positive or negative events which that issue causes in the ERP implementation project. Some examples of events are: a project activity is overdue, a lacking resource, an alert that the functionality of the new ERP system cannot support a certain part of an organisation or an angry email from a manager stating that his department no longer will participate in the project. These examples can be considered negative events. Of course also positive events exist, for instance: being ahead of schedule, or becoming aware that the functionality of the ERP software also can support other processes in the organisation.
Events are signs or symptoms of an underlying issue. For instance, if a project member calls in sick (an event!), the underlying issue could be a bad relationship between the project manager and project member. After this sick report, the project (in this case the project manager) could decide whether to accept this sick calling and assigning the work to another project member or to solve their relationship problems. Therefore as soon as an event arises, the project has to determine whether and how the event influences or may influence the project. In other words: what is the underlying issue which caused this event? As a consequence, an event always calls for one of more decisions, i.e. a decision has to be made whether to react within the project or not.
In general, as a consequence of one or more events, one or more decisions must be taken to solve the underlying issue which causes the events.
For instance, suppose an event occurs that a certain project activity is late because more work than planned is needed to be able to finish the activity. If this activity is an activity on the critical path, then the ERP project is in danger of running out of time. Running out of time will be considered an important issue, as it is a threat to the time goal set for the project. The event of the notification of a project activity being overdue, revealed this issue. One or more decisions have to be made as a result of this event to be able to solve this issue. The project could accept the fact that the activity is late and that the time goal of the project will not be met, in other words, change the project goal. Another solution could be finishing the project activity in time by applying overtime. One or more decision-making processes are gone through. In either case, the underlying issue, the danger of running out of time, needs decision making. Even if an event and issue are ignored, this can be considered a decision, explicitly or implicitly taken.
In case decisions have been taken which also call for actions, in most cases, these actions will be executed. For instance applying overtime, hiring more resources or summon the unwilling manager of the department to participate in the project. However, the responsible persons or organisations might not accept some decisions. This

nonacceptance could result in not performing the expected actions, performing other than the decided and expected actions, or not performing actions at all.
If the actions or lack of actions don't lead to a solution of an issue, the events may continue to exist, and/or even new events will arise. The issue is ***recurring*** and the chain of decision making, decisions and actions will be again gone through. Of course an issue also can be ignored, but still, ignoring is a decision-making process with the decision: not to perform any actions to solve the issue. These decisions might be unacceptable by stakeholders, in which case still actions are performed by these stakeholders leading to new events belonging to the same issue. Even if the decision not to take any action is acceptable by stakeholders, this still can lead to new events. The issue still exists and as defined below, is an actual or potential threat to the ERP implementation.
Also if an issue itself is unclear, there is a strong possibility that the decision taken and actions carried out are not the correct ones. If not the correct actions were taken, this could lead to more events by which an issue recurs.
If it is not clear whether the decision maker has the authority to decide, then it is possible that the decisions and resulting actions will not be accepted. In that case, an issue will not be solved, and the issue can generate new events.

Figure 5.1 shows the conceptual model of how we consider the handling of the processing of an issue in an ERP implementation project.

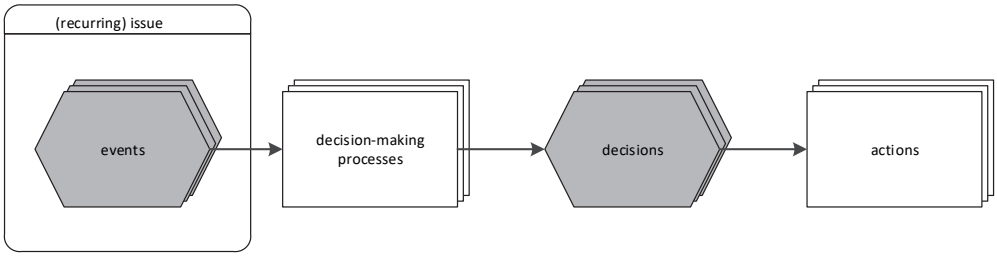


Figure 5.1 Issue handling model

To be able to detect issues in our research clearly, we have to define what we mean by the elements in the model shown in figure 5.1. These definitions enable us to detect and select issues in empirical cases.

Event	=	A signal triggering a decision-making process.
Issue	=	A perceived actual or potential threat to the goals set within an ERP implementation project.
Recurring issue	=	An issue for which events remain occurring.
Decision-making process	=	The clarification of the event and underlying issue, generation of possible solutions and the manner by which and by whom a decision is taken for handling the event and/or issue.
Decision	=	The selected action/actions.
Action	=	One or more activities which are carried, or not carried out as consequence of the decision.

A short **fictional** example of the components of this model:
During testing of the selected ERP software, the testers in Organization ABC become aware that the ERP system cannot support a specific business process. This insight emerged only during the testing of the software and was not known or expected when the ERP software was selected and the project planned. In this example, the problem report from the project manager to the steering committee is the ***event***. This report informs the steering committee that the ERP system cannot support that business process. A misfit of the ERP software in the organisation is the ***issue*** which caused the event. In the ***decision-making process,*** the steering committee studies the report, discusses possible solutions, like adapting the ERP system or changing business process X. Finally, the steering committee takes a ***decision*** that business process X should be changed to fit the ERP system. The project manager takes ***action*** by contacting the owner of business process X and explains the necessity for change. Unfortunately, the owner of business process X is not willing to change business process X and informs the project manager about his refusal by a memo, which in our model represents a new ***event***. The project still cannot continue because the misfit still exists. Therefore the ***issue***, being a misfit of the ERP software in the organisation, is not solved and can be considered a ***recurring issue***. This unwillingness impels the project manager to write another report to the steering committee, which is another ***event*** for the same issue. The steering committee starts another ***decision-making process*** and discusses possible solutions. Finally, the steering committee takes the ***decision*** that the ERP system will be adapted to be able to support business process X. The project manager takes ***action*** by letting programmers change the ERP system. The misfit is solved and therefore the ***issue*** is solved and causes no more ***events***.

5.2.3 Tracing unexpected issues

To be able to demonstrate the unexpected behaviour of ERP implementation projects we aim at showing evidence of unexpected issues in ERP implementation projects.

However, we expect that not all unexpected issues in an ERP implementation project can easily be retrieved. We presume that unexpected issues which a project or an organisation quickly and adequately handles are difficult to detect by research. These issues probably will not linger very long in the memory of the concerned persons or will be extensively referred to in documentation. Sometimes even unexpected issues can informally be solved even before they come consciously to mind to the stakeholders in an ERP implementation project. For instance, an informal conversation at a party between the project manager and a business manager about their project might already remove potential obstacles to the project and secure cooperation and acceptance.

Therefore in our research, we considered which subset of unexpected issues might best be objectively traceable in ERP implementation projects. Issues which are not solved right away are most likely to keep on causing events. We expect that participants in the ERP implementation project remember these events and that these events can be traced in documents. Therefore we assume our research might best trace recurring issues, which are not solved right away. For that reason, we will focus on detecting *recurring unexpected issues* as main evidence for the existence of unexpected issues in our case study.

5.2.4 Recurring unexpected issues

In the previous sections, we explained the model which we use as our research view on what issues and unexpected issues are, and how they are dealt with in an ERP implementation project. Also, we reasoned that detecting unexpected issues in research might be easiest by detecting *recurring unexpected issues*.
In our model, we defined a recurring issue as an issue for which events remain occurring. Therefore the recurring of an issue can be revealed by detecting events that keep on appearing belonging to that same issue. But when searching for recurring issues in implementation projects, what subset of issues should we focus on to find most likely issues that are recurring as well as unexpected? For this, we will focus on two characteristics of issues.
First, when events occur, and the underlying issue which causes these events is unclear to the ERP project, there is a high probability that ineffective actions are taken for that issue. These ineffective actions might make it difficult or impossible to solve an issue. The issue might not be solved, and events might keep on occurring. On the other hand, if events occur and the underlying issue is clear to the ERP project, we expect a higher probability that proper actions are taken for that issue, and the issue might be solved. Therefore we differentiate issues in issues that are clear and issues that are unclear to the ERP implementation project.
Second, if the authority for solving the issue is unclear or missing, it means that the ERP implementation project is not designed and properly set up with the authority or access to the proper authority for solving that type of issue. If it was expected that issues of a certain type could come into existence in the ERP implementation project, the project would have been equipped with the proper authority for handling that type of issue. Also if the authority for solving an issue is unclear or missing, then despite clearness of the proper actions to solve the issue, these actions may not be executed as there is no adequate consultation body where the issue is discussed and decided upon. As a consequence the project itself, due to lack of authority, cannot execute the necessary actions, which might lead to new events. As previously described, we consider an issue (recurring or not) unexpected if the authority for solving the issue is unclear or missing. Therefore we also differentiate issues in issues for which the decision authority is clear and present, and issues for which the decision authority is unclear of missing.

If we differentiate issues by these two characteristics: the clearness of the issue and the clearness and presence of the decision authority, then we can distinguish four types of issues as shown in table 5.1. In table 5.1 we also indicated for every type whether it could be expected that that type of issue will keep on causing events.

Table 5.1 Types of issues

	decision authority clear and present (inside or outside the project boundaries)	**decision authority unclear or missing**
issue clear	Type A No additional events might keep on coming.	Type C Events might keep on coming and/or are ignored (hushed up).
issue unclear	Type B Events might keep on coming until the issue is clarified and the proper solution is applied.	Type D Events might keep on coming and/or are ignored (hushed up).

We labelled these issues type A, B, C and D. Next we will discuss these types of issues.

Issue type A
In an ideal situation, an issue is clear, and the authority for deciding on taking actions to solve that issue is clearly known. In that case, we expect a high probability that an issue will be solved. Even if the authority for solving an issue is outside of the project, if it is clear what an issue is and who is authorised to decide on taking actions, an issue might be solved without recurrence.

Issue type B
If an issue is unclear, but the authority for deciding on taking actions is clear, then we expect that an issue will not be solved right away and keeps on recurring. Although the correct decision-maker will decide on actions, there is a possibility that these are improper decisions due to the unclearness of the issue.
However, when more events occur related to the same issue, an issue might become clear in a way that appropriate decisions can be made and proper actions performed.

Issue type C
In a situation where an issue is clear, but the authority for deciding on taking actions is unclear, an issue might also not be solved. In that case, a project might mistakenly assume that someone within that project has the authority to solve the problem. Also,

if a project is aware it does not have the authority to solve the problem, but it is unclear where in an organisation this authority resides, an issue probably will not be solved and will recur.
For instance, it could be unclear whether a specific manager in a department has the authority, or whether an issue needs a decision from the board of directors. Another type of C issue could be that a project assumes to be authorised for an issue, as the authorization was described in its project definition, whereas in reality outside of that project that authority is unaccepted.

Issue type D
In the worst of situations, an issue is not clear and also the authority for deciding on taking actions is not clear. The latter might also be caused by the fact that an issue is not clear. In that case, it is obvious that there is a very high probability that an issue will not be solved and will recur.

By reasoning according to this model, we assume that the possibility of finding in actual ERP implementation projects multiple events belonging to the same (recurring) issue is highest for issues type B, C and D. However, as we have an aggregate complexity viewpoint in mind, we not only aim at retrieving recurring issues but aim at retrieving unexpected issues. We reasoned before that type A issues might also be unexpected, but there is high chance that this type of issue will be solved. This because the issue is clear and the decision authority is clear. We expect that these issues are difficult to detect in empirical research. Type B issues can also be expected to recur but may be solved in the end by the project itself, as the project has the authority to solve this type of issues.
Type C and D are the types of issues which we expect to generate multiple events and also due to unclearness of decision authority can be considered unexpected issues. Therefore in our empirical research, we will focus on issues in ERP implementations for which the project also had an unclear or missing decision authority. Which means that we will focus on detecting C or D types of issues, which we expect are the easiest to detect and also can be considered unexpected issues.

Table 5.2 shows the type of issues we intend to retrieve empirical research.

Table 5.1 Types of issues to retrieve from empirical research

	decision authority clear	**decision authority unclear**
issue clear		Type C
issue unclear		Type D

5.3 Research design

5.3.1 Introduction

In this section, we will discuss the considerations which have led to the design of our empirical study by an aggregate complexity approach. We performed an intensive case study into an ERP implementation project from a public body. In this case study, we gathered information from project participants and relevant documents in several steps. Arguing from this information we tried to determine whether this ERP implementation project had unexpected recurring issues.

5.3.2 Research strategy

To be able to perform some exploratory research into whether unexpected behaviour in ERP projects can exist, i.e. unexpected issues occur despite correct project preparation, planning and project management; there are several options to be considered.

In an ideal situation, an intensive objective observation and documentation of the planning and progress of a large collection of ERP implementation projects would be a solid strategy to detect these unexpected issues. In that case, every event in these ERP implementation projects could be observed and analysed with all information still present and accessible. However considering the long durations of these types of projects, the difficulty of getting access to these projects and also the effort needed to observe and document a large collection of ERP implementation projects, this seems an unpractical and unrealistic research strategy.
Another alternative could be to select a large representative collection of ERP experts and survey this group about the existence of unexpected issues in ERP projects. Although this could cover a large collection of relevant ERP projects and would be practical regarding the duration of the research, the outcome of a survey would be a collection of opinions of experts and unfortunately not well-founded objective observations.

Our research question for this section is: to perform some exploratory research to discover the existence of unexpected behaviour by demonstrating unexpected issues that are clearly out-of-scope of the implementation project and can only be solved outside of the project itself. Therefore, theoretically finding several of these issues in one carefully selected case could be an indication. Although also in observing one characteristic ERP implementation project, the duration would exceed the possibilities of available research time. Therefore we considered a thorough analysis of a completed characteristic ERP implementation project, by purposeful sampling (Coyne, 1997) an appropriate choice, bearing in mind the objectiveness and depth of the results compared to collecting opinions of experts.

Also, this part of our research has an exploratory purpose into whether the aggregate complexity research approach can be supported by evidence in practice.

5.3.3 Case selection

We consider it not feasible to determine pure objective and quantitative criteria for selection of an appropriate case. Using objective criteria like number of employees, number of project members, number of ERP modules et cetera, can hardly mutually compare the complexity of ERP projects. As stated before, complexity is not only dependent on the structure of a system, but also depends on influences from the surroundings and behaviour of its subsystems. These structures and external influences will be unique in every ERP project. Therefore selecting an appropriate case by using merely quantitative and objective criteria, is unrealistic and disregards the unique complexity of every ERP project.
Therefore we believe that a case can only be selected based on qualitative characteristics.

Hence for our appropriate research case we aimed at a case with the following qualitative characteristics:

- The ERP implementation should have taken place in a professional organisation with a professional project management organisation and skilled and experienced project managers. It is obvious that in an organisation where projects are performed without a professional project management organisation, unexpected issues can arise. These issues than cannot be linked to the complexity of ERP projects.
- The project, as ERP projects mainly are, should be considered an important, large and costly project. Important and large projects will be seriously managed and controlled within the organisation.
- The ERP implementations should have taken place no longer than three years ago. The memory of what has happened and also relevant documentation should still be present.
- The organisation should be willing to provide all access necessary for our research activities and supply information in an open and honest way, without any restraints. If there is no access to all information, the outcomes of the research will be unreliable. Our case research intends to detect and discuss issues as its main objective. Most organisations do not allow research into projects they succeeded or even worse not-succeeded. Unfortunately, most organisations do not like to reveal their problems to outsiders and even less likely allow publication of these problems.

5.3.4 Case information sources

For our research, we intended to retrieve detailed and explanatory information about issues which arose during the ERP implementation project. Whether something was or should be regarded as an issue, can only be indicated by humans. Therefore this will be our primary information source. However, also documents which hold information can be used to check the information retrieved from humans.
Other information resources like media and literature are less likely to contain any information about a particular ERP implementation. In general, media do not discuss ERP projects. Also, academic publications do not contain in great detail elaborated case descriptions.

For a selected ERP implementation case the next information sources should be relevant and available to obtain information about issues:

- Stakeholders with knowledge of and experience from that particular ERP implementation. This information source can provide facts and opinions about the finished ERP project. Although it will be dependent on the case organisation and project organisation which type of stakeholders will have the most relevant knowledge of the project, it can be expected that project managers, general managers, consultants, (super)users and project members are stakeholder types which can provide information.
- Documents which hold information about this particular ERP implementation: Project definition reports, management reports and other relevant reports dealing with the ERP implementation project. These documents could provide information about issues for triangulation.

5.3.5 Main research steps

To retrieve information from the ERP project participants and documentation, we divided this case study into several steps. As it would be unfeasible (by influencing) to ask respondents directly what issues of type C and D came up during the ERP implementation project, a more objective approach was designed. The goal of this approach was not to reveal to the organisation what types of issues we were trying to detect. Also, not to reveal why we were searching for these types of issues, as this might create bias in the organisation. Directly asking for this type of issues, on the one hand, could cause twisting facts to prove in such a way that every issue that occurred never was caused by poor project management or human error. On the other hand, it could cause the opposite, by assigning a culprit as an explanation of the cause of every issue, and to state that this will never occur in future again (the deterministic paradigm) by using better experts or methods.
To avoid these unwanted biases of the participants in a case, we intended to search and classify the unexpected issues in the best possible objective manner. We tried to

discover the existence of unexpected issues, only based on the discussed model of issue handling and the discussed classification of the issue types. We used the information retrieved from the employees and documentation and not by classification by the case organisation itself.
Therefore we first collected as much information as possible about that ERP implementation project and its subprojects through documents. We analysed these documents with the definitions of issues in mind to retrieve issues which arose during the project. Next, we intended to retrieve information about issues from the participants from the project and finally compose a list of issues from that project. Every type of issue could be on that list. To retrieve information about issues of the participants, we chose to use semi-structured interviews. As for reasons for avoiding bias, we could not ask directly by structured interviews after type C and D issues. Also, we needed a thorough explanation of the background of every issue, which would not be feasible with a fully structured interview. With our definition of type C and D issues in mind, we further intended to explore the course and background information for every issue by documents and participants. At the end with that information, we should be able to argue which issues in the case were of the unexpected type we were looking for.
In the next section, we will explain our planned research steps for our case study in detail. Figure 5.2 shows a diagram of our research steps.

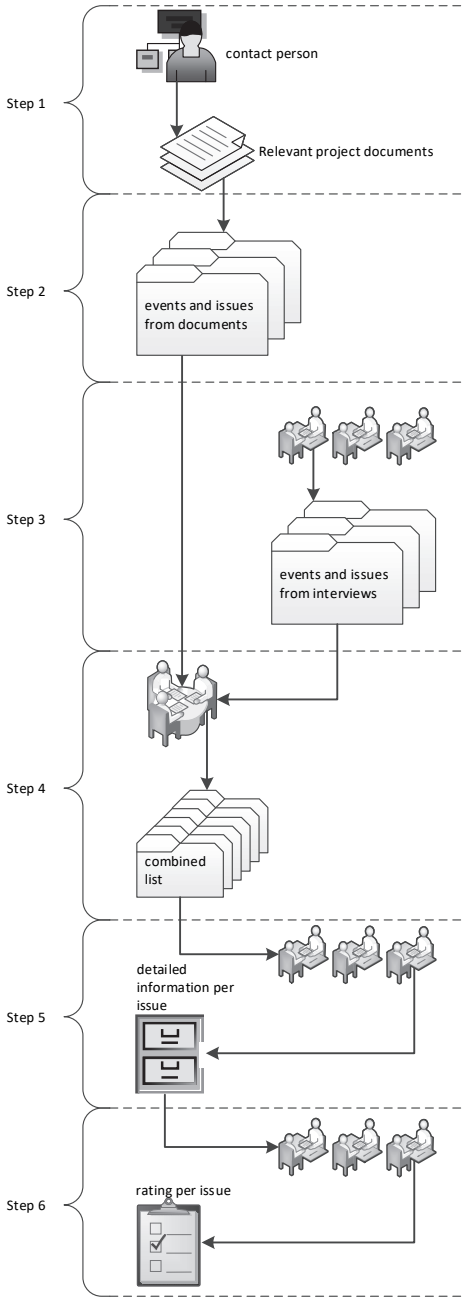


Figure 5.2 Diagram of research steps

Step 1: Gain access to a contact person and assuring confidentiality
After selection and admittance to the case, in this step a main contact person within the case should be requested. The function of this contact person should be to provide assistance in gaining access to relevant documents and project participants. Also, this contact person could support in arranging appointments and assist in practical matters, for instance, access to the internet when on site, or arranging rooms for interviews or presentations. This contact person should be well informed about the ERP project and have access to all managers, project participants and documents.
Step 2: Extraction of events and issues from project documentation
The researcher should gain access to project definition documents, steering reports, project progress reports and, in the opinion of the contact person, other relevant documents.
By studying the content of the provided documents, the researcher should evaluate these for potential events and issues. The researcher should interpret the documents with the discussed research definition for events and issues in mind, and create a list of retrieved events/issues.
This list should be revised and validated by the contact person from the case organisation.
It is important to note that this list of events/issues is merely a minimal list and not the complete list of issues/events in this ERP implementation project case, for the following reasons:

1. Not all events/issues may have been recorded in the documentation.
2. Not all documentation may have been retrieved or made available.
3. Not all events/issues may have been detected by the researcher in the documentation.

Figure 5.3 demonstrates this graphically.

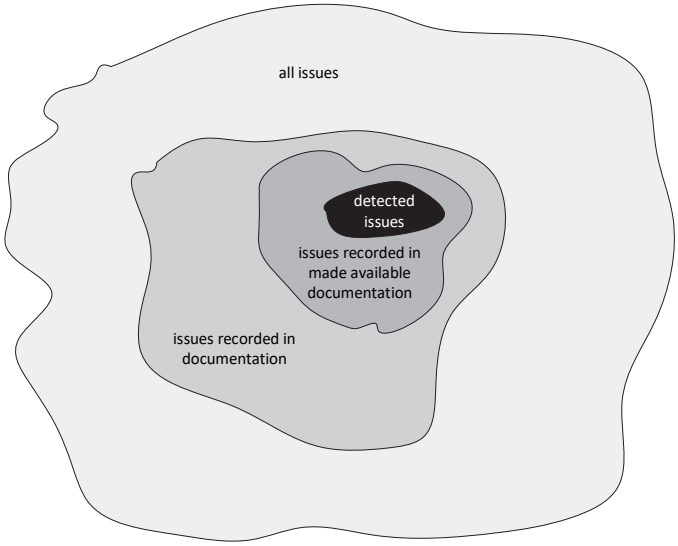


Figure 5.3 Limited subset of detected issues

Completeness of all events/issues is not the purpose of this step, but the establishment of a worthwhile list of potential recurring issues for further deepening in the next steps.
For this reason interpretation of the documents by the researcher and revision and supplementing by the contact person should efficiently and effectively deliver the results.
Next, the researcher should determine in consultation with the contact person which ERP implementation participants should be interviewed in the next step. Also, the structure and content of the interviews should be designed.

Step 3: Extraction events and issues by interviews
After step 2 the researcher should have a comprehensive overview of the project and should have retrieved as much events/issues from the documentation as possible. In step 3 confirmation of this list and additional issues/events should be obtained from ERP implementation participants.
As already indicated in step 2, not all events/issues may have been recorded in documentation. Therefore step 3 will serve two purposes:

1. Discovery of events/issues not documented or missed in the documentation.
2. Confirmation of events/issues already discovered in step 2.

The purpose of step 3 is similar to the purpose of step 2, i.e. retrieval of as many issues/events as possible in the most efficient and effective way. Therefore not all mem-

bers of the ERP project are needed to provide information, which would be unfeasible. Relevant participants will be selected with an extensive overview of the project. These persons should be selected in consultation with the contact person.

Step 4: Integration of events/issue lists
In step 2 and 3 events/issues should be recorded in separate lists. These lists need to be combined into one list, which can be considered a fair representation of events/issues from the case. To design a reliable list of issues, the researcher should perform this combining. The researcher's two supervisors and the contact person of the case should check the list.

Step 5: Profound potential C/D issue information retrieval
The issues from the last step should be analysed by interviews with relevant project members.
The information from the interviews and information from the documents should be combined in a structured file per issue. Every file describes the issue by:

1. Background of issue
2. Description of issue and events
3. Description of settlement issue by decision-making processes, decisions, actions and whether the issue was solved or not.

This file should contain all the elements from the 'Issue and event handling' model.

Step 6: Issue rating
Based on the gathered detailed information about the issues in the file from step 5, the case organisation perform and verify the definitive classification of the issues in A, B, C or D.

5.3.6 Validity and reliability of the research design

Before we discuss the results of our exploratory research by the aggregate complexity research approach into ERP implementations, we will discuss the validity and reliability of our designed case research method.

Construct validity
The construct validity of a research procedure refers to the extent to which a study investigates what it claims to investigate, that is, to the extent to which a procedure leads to an accurate observation of reality (Gibbert & Ruigrok, 2010).
In this case study, we tried to assure the construct validity by several actions.

First, we designed the issue and event handling model (section 5.2.2) to clarify and anchor the subject we were trying to detect: in our case recurring unexpected issues in an ERP implementation project.
Second, we tried to assure the construct validity by triangulation of the results from different sources of data. To obtain possible issues we aimed at searching through documents in a structured way as described in step 2. After that, we turned to people as an information source for discovery of additional issues or events. As shown in the previous discussion of the research steps, on the way we should combine the results and secure the results by as much information sources as possible.
Third, we tried to assure a high construct validity by discussing in detail all the steps taken to obtain and analyse the data and documenting these steps.

Internal Validity
Internal validity or ''logical validity'' refers to the presence of causal relationships between variables and results. Gibbert and Ruigrok (2010) discuss that the single main challenge for qualitative researchers wishing to ensure validity is to convince themselves (and their audience) that their findings are genuinely based on a critical investigation of all their data and do not depend on a few well-chosen examples.
We used triangulation in our research design to assure that. We analysed all available ERP project related documents and inquired within the interviews after all issues from the project. As we should not reveal the purpose of our research to the interviewees, we should not specifically ask for recurring issues, but we should ask for issues/problems in general. For every potential recurring issue, we should try to detect objectively as possible (by triangulation) whether it is an issue which unexpectedly occurred. By discussing every potential recurring issue in separate interviews in depth and documenting and verifying the results of these interviews, we should try to transparent underpin the conclusions, i.e. whether unexpected issues exist in this ERP project.

External Validity
Gibbert and Ruigrok (2010) discuss that ''External validity,'' or ''generalizability'' is grounded in the intuitive belief that theories must be shown to account for phenomena not only in the setting in which they are studied but also in other settings.
We carefully select a case by purposeful sampling (Coyne, 1997). We intend to select our case very carefully to be able to apply our findings to other comparable situations. But of course, we are aware that our results might not apply to situations which considerable deviate from our selected case type. In that case, more research into these deviating types should be performed. However, Gibber and Ruigrok also mention that conducting different case studies within one organisation might also improve the external validity of qualitative research. Therefore we aimed at selecting an ERP implementation which consists of several sub-projects, with their specific project leaders, steering committees, scopes et cetera.

Reliability

In this study, we tried to ensure the reliability by the next choices and actions.
To improve a correct representation of the information, we should retrieve from the interviewees; we should tape every interview. Based on this recording, we carefully should make a resume for the interviewee, which he or she confirmed and/or can enhance. An important aspect of reliability is transparency of the research protocol (Gibbert & Ruigrok, 2010). In section 5.3.6 our research steps are discussed in detail and also the results from every step must be available (except the recordings of the interviews as a result of confidentiality) through the author.
Another important aspect of reliability is replication. Although of course interviews cannot be repeated in the same way, the elaborated results can be used to replicate assigning the type of issue (A, B, C, D). Also from documents retrieval of possible issues can be replicated and triangulated with results from interviews.
However, we are aware that assigning the type A, B, C or D to an issue is a judgement. By assigning the type of an issue by several well-informed persons, we expected that this triangulation enhanced the reliability.
Of course, we are aware of the fact that researcher's bias can always influence the results of the interviews. Although we tried to avoid questions by which the purpose of our research could be revealed, the researcher's bias might unconsciously influence the questions asked during the semi-structured interviews and recorded results. E.g. unknowingly asking and recording what the researcher would like or expected to find.

5.4 Research results

5.4.1 Case description

In the previous section, we described the main characteristics of the required research case: a large and important, professionally managed, recent and accessible ERP implementation project. We were very fortunate to be able to obtain approval and cooperation from a large public body in the Netherlands. This public body had implemented and still was implementing ERP by a professional project organisation. This implementation affected large parts of the public body itself.

This public body consists of over 19,000 employees, who mainly perform service tasks to the Dutch population.
The public body has a professional IT organisation and is familiar with large and complex IT projects. Their proprietary systems and websites are supporting their provided public service.

Although the ERP software does not support the primary public processes of this public body, it supports the supportive processes like human resource management, procurement, financial management and IT. Nevertheless, the implementation of the ERP software affected all 19,000 employees. For instance, the ERP software supports the administration of and requesting leave, and the ordering of office supplies.
This public body already used Oracle's PeopleSoft[4] ERP software for financial management. Still, it started in 2013 with projects for implementing more of PeopleSoft applications. The goal was to phase out several expensive legacy systems and create an integrated, cost and support efficient information system.
To achieve this, the public body carefully planned and executed projects which implemented parts of PeopleSoft. They used a piecemeal approach to implement applications of PeopleSoft. The main subprojects in this ERP implementation were:

1. Support for procurement of goods and services for Financials and Supply Chain Management.
2. Registration and control of absence and leave for all 19,000 employees.
3. Support for procurement of goods for a public service of the public body.
4. Integration of the use of the business card for public transport in the Netherlands into the financial employee administration.
5. Course administration for own employees due to legal requirements (for instance for physicians)
6. Support for handling administrative tasks on entering new employees

[4] http://www.oracle.com/us/products/applications/peoplesoft-enterprise/overview/index.html visited October 15, 2016

7. Integration of information systems of local authorities with the ERP system of the public body.

The public body had an own department which is dedicated to the implementation and support of ERP systems. This department contained several experienced IT project managers. Also, the public body had a program department were every IT related project and subproject was assessed and monitored. Project management was performed in accordance to and by the standards of the PRINCE2[5] methodology. The project managers had to be PRINCE2 licensed.

Also, they contracted a consultancy firm for parts of the ERP implementation to advise and assist where they lacked knowledge and experience with PeopleSoft applications. The project managers were own employees. The general managers of the staff services carefully initiated and supported this ERP implementation.
Considering all these characteristics of the project organisation, project managers profiles and project management standards, we supposed that the case organisation had a professional project organisation and worked according to professional standards, which satisfied the qualitative requirements stated in the previous section.

After a formal request for performing research after this ERP implementation, the public body granted our request. The public body cooperated in our research in every way. The organisation provided project documents and access to project managers, managers and members of the project to obtain the necessary information. However, because the case organisation is a public body, the researcher had to assure confidentiality of the obtained information.

The identity of and further information about this public body can be obtained from the author, who of course always will need to ask permission to the public body before releasing this information.

5.4.2 Results from main research steps

To retrieve information from the ERP project participants and documentation, the next steps were performed and results obtained:

Step 1: Gain access to a contact person and assuring confidentiality
A contact person within the public body was requested and granted. This contact person was well informed about the ERP project and had access to all managers, project participants and documents.
Also as a requirement from the organisation, a standard confidentiality statement for employees was signed by the researcher.

[5] https://en.wikipedia.org/wiki/PRINCE2 visited October 15, 2016

Step 1 was performed as planned.

Results

- Access to the case by an adequate contact person
- A signed confidentiality statement

Step 2: Extraction of events and issues from project documentation
The researcher requested the contact person to provide access to project definition documents, steering reports, project progress reports and, in the opinion of the contact person, other relevant documents. In total, the researcher gained access to 129 documents, which were all electronic files.

The researcher evaluated the content of these 129 provided documents for potential events and issues. The researcher interpreted the documents with the discussed research definition for events and issues in mind and created a list of retrieved events/issues. As the documents were all in electronic format, the researcher highlighted in every document sentences which could indicate an issue or event. These sentences were collected in an Excel file with reference to the document.
In the end, the researcher retrieved 72 possible events or issues.

This list was revised and validated by the contact person from the case organisation.

Next, the researcher in consultation with the contact person determined which ERP implementation participants he should be interviewing in the next step. Also, he designed the structure and content of the interviews.
Also, step 2 was performed as planned.

Results

- 129 Relevant ERP implementation documents
- Understanding of the project by the researcher
- A first list of 72 issues and events extracted from relevant documents
- A review of interview questions and approach
- A list of four names for interviewing in the next step

Information sources

- Documentation relevant to the selected ERP implementation project
- Contact person from the case organisation

Step 3: Extraction events and issues by interviews
After step 2 the researcher had a comprehensive overview of the project and had retrieved as much events/issues from the documentation as possible.

The purpose of step 3 was similar to the purpose of step 2, i.e. retrieval of as many issues/events as possible in the most efficient and effective way. Therefore not all members of the ERP project needed to provide information, which also would be unfeasible and unacceptable by the case organisation. Therefore four participants were selected with an extensive overview of the project. These persons were selected in consultation with the contact person. These four participants held at that time the following positions (job titles) within the organisation:

1. Director of facility management and purchasing
2. Manager ERP competence center
3. Manager project professionals
4. Project manager ERP

All participants had from two to over ten years' experience with ERP implementation projects and were at least two years involved in the ERP implementation at this public body.

The researcher retrieved by semi-structured interviews the events/issues from these four selected persons. Semi-structured face-to-face interviewing had, in this case, the advantage of being flexible in exploring the experiences of the interviewees in the project. A written questionnaire would not be flexible enough to perform this task, for the same reason as a structured interview would be too limiting. It would be necessary to explain to the interviewees what this research considers an event/issue. The interviewees could interpret this explanation differently. By using semi-structured interviews, the interviewer could adjust the questions and react, if necessary, to the answers with the concepts of event and issue in mind.
Another reason for using semi-structured interviews is the necessity of avoiding bias when asking for issues/events. It was necessary to allow the interviewees as much freedom as possible in describing what had happened during the project, without revealing the purpose of this research. Questions such as who had the authority to decide upon an issue, or who did decide on an issue, which is information we ultimately needed to know, would put bias on the collection of events/issues and possibly reduce frankness. Therefore only a few questions were put forward to get the mental process started for the interviewee for remembering the project. The interviewer i.e. researcher did not influence the interviewee but merely supported the interviewee in remembering issues/events by asking supportive and explanatory questions.

Once the interviewee indicated that he could not recall more issues and events, the interviewer discussed the list from step 2 with the interviewee. This discussion was done with the purpose to trigger remembering of more issues and events and also to obtain a confirmation of the issues and events from the documents. Appendix 5.1 shows the interview manual (in Dutch) used in step 3.

By these four interviews, 42 issues/events were captured.

Every interview was recorded by a voice recorder. After elaboration of the recording by the interviewer, a few days later the interviewer presented a list of events/issues with a short description per event/issue to the interviewee for confirmation and possible enhancements. After confirmation and enhancement by the interviewee, the recording of the interview was deleted for reasons of confidentiality, which had previously been agreed.
Step 3 was also performed as planned.

Results

- 42 discovered events/issues from the interviewees.

Information sources

- Participants of the ERP implementation project who were well aware of the entire project. Quantity: four interviewees.
- The list of issues from the documents from the previous step.

Step 4: Integration of events/issue lists
In step 2 and 3 events/issues were recorded into five separate lists. These lists needed to be combined into one list, which could be considered a fair representation of events/issues from the case. The researcher composed a first combined list. This list was checked by the researcher's two supervisors, after which the result finally was checked by the contact person from the case organisation.
Step 4 was performed as planned.

Results

- An integrated list of 114 discovered events/issues.

Information sources

- List of events/issues from documentation
- Lists of events/issues from four interviews

- Contact person
- Supervisors

Step 5: Profound potential C/D issue information retrieval
Initially, it was intended to analyse in step 5 every retrieved issue in detail to be able to indicate which issues were from C or D type. However, the list of 114 discovered issues was too big. It would have been unfeasible and also not acceptable by the case organisation to analyse all 114 issues in detail. This analysis would imply discussing every issue on this list with several interviewees. At this stage of the research, unambiguous classification of issues into A, B, C and D was not yet possible due to insufficient information. Therefore a research redesign decision was made to add an intermediate step. In this intermediate step, potential C and D candidate issues within these 114 issues would be selected by the available knowledge of the case. The researcher made this selection and discussed this selection with his supervisors. Finally, the researcher discussed and verified this selection with the contact person, whom he only now also revealed the purpose of this case research. However, to be able to retrieve unbiased results in the next steps he requested the contact person not to reveal this purpose in the case organisation.
Finally, in this step 11 potential C or D issues were selected for detailed analysis.

Also, the researcher determined in consultation with the contact person which ERP implementation participants the researcher next should be interviewing for these 11 issues. Also, the structure and content of the interviews were designed. Appendix 5.2 shows the interview manual (in Dutch) used in step 6.

The selected 11 potential C/D issues were analysed by four interviews to be able to classify in the next step the issue as a C/D issue or as an A/B issue definitively. The contact person in consultation with the researcher selected four interviewees which should be able to provide detailed information about the issues. Furthermore, to limit the burden on the organisation and protect the willingness to cooperate, individuals were selected who could provide information about multiple issues. These individuals were selected in a manner that at least two individuals could provide information about every issue. The necessary information for every issue was retrieved from the selected persons by semi-structured interviews. For the same reasons as the use of semi-structured interviews in step 3.
Similar to step 3 every interview was recorded. After elaboration of the recording by the interviewer, the information provided about the issues was presented to the interviewee for confirmation and possible enhancements.

The four participants held the following positions (job titles) within the organisation:

1. ERP consultant
2. Manager ERP strategy and development
3. Senior project manager
4. Project manager ERP

The information from the interviews and information from the documents was combined in a structured file per issue. Every file described the issue by:

1. Background of issue
2. Description of *issue* and *events*
3. Description of settlement issue by *decision-making processes*, *decisions*, *actions* and whether the issue was solved or not.

This file contains all the elements from the Issue and event handling model.

Purpose
Profound information retrieval of the issues.

Result

- A list of 11 **potential** C/D issues.
- For each **potential** C/D issue an interpretation of the individual components of the issue model.

Information sources

- Per **potential** C/D issue, at least two people who were well aware of the issue and did describe the issue from a different point of view.
- Project documentation (e.g. steering committee reports, project diary, etc.).
- Other relevant documentation (e.g. management reports etc.)

Step 6: Issue rating
Based on the gathered detailed information about the 11 issues in the file from step 5, the definitive classification of the potential C/D issues in A, B, C or D was performed. The researcher first performed the classification based on the information from step 5. After that, the researcher's supervisors controlled and confirmed the classification. The classification was also performed independently by the contact person. As one of the interviewees was a former consultant of the public body who advised during the implementation and also had a complete overview of the project, this consultant was also invited to rate the 11 issues.

As shown in table 5.3, an issue was only classified as an C/D type issue if all rated the issue as of C/D type.

Table 5.3 Final rating of issues (C/D issue: Yes of No)

Issue	Researcher and supervisors	Former consultant	Contact person	Final rating
1	Y	Y	Y	*Y*
2	Y	Y	Y	*Y*
3	Y	Y	Y	*Y*
4	Y	Y	Y	*Y*
5	N	N	Y	*N*
6	Y	Y	Y	*Y*
7	Y	Y	Y	*Y*
8	N	N	Y	*N*
9	Y	Y	Y	*Y*
10	N	Y	Y	*N*
11	N	N	Y	*N*

Purpose

- The rating of the 11 potential C/D issues in the issue matrix is performed based on the issue analysis file and rating of two employees of the public body.

Result

- A founded list of seven **verified** C/D issues

Information sources

- Issue analysis files from previous step
- Contact person, supervisors and former ERP consultant

Step 7: Case organisation assessment on unexpected issues
As the case organisation felt it was important to reveal the results of the research to the department which supported the ERP implementation, a presentation was scheduled for this department. Although this was no planned step in the original research plan, we considered this an extra opportunity to validate the conclusions from our research. Moreover, to gain a first indication whether the aggregate complexity research approach could be useful for practice.

Therefore during this meeting the researcher was able to ask approximately 20 representatives of this department about their opinion on the existence of unexpected issues in well planned and managed ERP projects. Also, he was able to ask their opinions on the usefulness of the aggregate complexity research approach for ERP implementations.
In that meeting, we presented the results to the 20 attendees. However, the detected issues were not discussed in detail. We intended to avoid in the meeting an everlasting discussion about what exactly the reason for the emergence of an issue was. Also, the case organisation requested in advance not to discuss the detected issues as to avoid designation of culprits. In this meeting only our overall conclusion, stating that unexpected issues in their ERP project existed, was presented. Furthermore, the revelation of the purpose of the research, the performed research activities and the discussion of the complexity paradigms were presented. At the end of his presentation, the researcher specifically asked whether the attendees agreed that unexpected issues during their ERP implementation project arose. From that discussion and remarks the attendees made, it can be concluded that also in their opinion unexpected issue arose. These remarks were noted during the discussion. Appendix 5.3 shows the comments from the case organisation during the presentation of research results.

Purpose

- An opinions based confirmation of the conclusions that unexpected issue exist/existed in the ERP implementation project by the case organisation.
- An opinions based confirmation whether the awareness of aggregate complexity can support and enhance managing ERP implementations.

Result

- The insight that project managers get frustrated because the occurrence of unexpected issues is often considered as caused by the incompetence of project managers.
- A confirmation of the assumption that unexpected issues exist/existed in the case ERP implementation project.
- A confirmation of the assumption that the awareness of aggregate complexity can support and enhance managing ERP implementations.

Information sources

- Interviewees
- Members of the ERP implementation department

5.4.3 Validity and reliability of the research results

Before we discuss the conclusions of our exploratory research into the aggregate complexity research approach for ERP implementations, we will discuss the validity and reliability of the results.

Construct Validity
We were able to perform our research along our designed steps.;
In our performed research we were able to use triangulation of the results from the provided project documents and the results from the interviews. Also, we checked with the contact person, who was informed with the exact definition of what a recurring issue is, whether the retrieved list of issues met that definition. For every step, we carefully documented the results. Because of the confidentiality of the data, not all data could be published in this thesis. Nevertheless, every result and data described in the research steps are available through the author of this thesis after permission of the case organisation.

Internal Validity
During our case study, we did not get any indication that we had no access to or received access to documents or people that held important information about the ERP implementation. Nevertheless, as we are not a part of the organisation, we never can be sure that no information knowingly or unknowingly was withheld.
To secure the internal validity as much as possible, we did not reveal our research purpose to the interviewees and only asked for issues/problems in general. However, at some point in time, we had to reveal our research purpose to the contact person. Although we urged the contact person not to reveal this purpose in the organisation, we could not verify whether the contact person complied.
We tried to enhance internal validity by first retrieving all issues (A, B, C and D issues) and only after applying the definition of recurring issues to it; we typified an issue as relevant or not. This typifying was also done by our contact person and by a special interviewee, after revealing for which issues we were looking. By documenting the results from our analysis of the project documents and the summary of our interviews, we expect the conclusions about our results can be verified and repeated.

External Validity
In our researched ERP implementation case the project consisted of several sub-projects, with their specific project leader, steering committee, scope et cetera. Our results show that there are recurring issues within several of these sub-projects, which enhances the external validity of our results.

Reliability
In our research, we tried to ensure the reliability by the next choices and actions.
To improve a correct representation of the information we retrieved from the interviewees, we taped every interview. Based on this recording, we carefully made a resume for the interviewee, which he or she confirmed and/or enhanced. Also, the by the researcher selected issues from the documentation were confirmed by the contact person as being a representative list of issues belonging to the ERP implementation project. An important aspect of reliability is transparency of the research protocol. Therefore we discussed the research steps in section 5.3.5 in detail. Furthermore, the results from every step are available (except of course the recordings of the interviews) through the author.
Of course, we would have liked to be able to interview more representatives from the ERP implementation project. We were not able to interview sheer users of the ERP system or retrieve documents from the user departments. As we were dependent on the contacts of the contact person and also the willingness of the organisation to cooperate, we had to compromise between the perfect information sources and available information sources. This compromise might have influenced the results. On the one hand, the compromise might have had its influence on not detected issues from the project, and on the other hand on the rating of the issues as C/D types.
Another important aspect of reliability is replication. Although of course the interviews cannot be repeated in the same way, the elaborated results can be used to replicate assigning the type of issue (A, B, C, D). Also, from the documents the retrieval of possible issues can be replicated and triangulated with the results from the interviews.

5.5 Conclusions and discussion for Study 4

5.5.1 Conclusions

In Study 4 we performed a case study to detect unexpected behaviour of an ERP implementation. Unexpected, despite a professional project organisation and careful preparation of the ERP implementation project.
Our goal in this study was to perform exploratory research to demonstrate the existence of unexpected issues that are clearly out-of-scope of an ERP implementation project and can only be solved outside of the project itself.
Unexpected behaviour in this study was detected by demonstrating that unexpected issues arose during that ERP implementation project. Issues which, despite careful planning and preparation, could not have been foreseen and therefore also could not have been taken into account when designing the project. These unexpected issues were detected by systematically searching for issues for which events kept on occurring and for which the authority of solving the issue was not clear or outside of the project. If such an issue could have been expected, this authority would have been considered in the design of the project.

As shown in table 5.3, we detected seven unexpected issues in this ERP implementation project. Seven issues that had a significant impact. By our access to the project documents, a professional organisation for support of projects, use of PRINCE2 as a project-management methodology, information from the interviews and general consultations with the case organisation, we consider that this organisation carefully prepared and managed this project and its subprojects by means of professional project management. Therefore we conclude that these seven issues can be considered unexpected and conclude that this case showed unexpected behaviour of an ERP implementation, despite a professional project organisation and careful preparation of the ERP implementation project. If this case can be considered an appropriate case for ERP implementations, for which this case was selected, we expect that also other ERP implementations can show unexpected behaviour, despite professional project organisation and careful preparation. If this is the case, research, as well as practice, should be aware of this and consider or design management techniques for managing ERP implementations in addition to ordinarily used project management.

5.5.2 Discussion

Are these seven detected issues sufficient support for the assumption that unexpected events will occur despite proper planning and management of ERP implementations? On the one hand, proof of just one unexpected issue would already proof that unexpected behaviour exists in ERP implementations, analogous to the evidence that black swans exist by finding just one black swan. On the other hand, it is difficult to justify that an issue within an ERP implementation project came into being despite a perfect

project design, preparation and management. Proof of a perfect project design, preparation and management are infeasible. Perhaps the preparation or management was not of adequate quality, which as a result caused the issue, as it is a human characteristic not wanting to admit mistakes.

However, if we assume that this ERP implementation project was adequate designed and performed, we detected seven issues which by an indication of multiple respondents and documentation research, could not have been prevented by better preparation or management of the project as current knowledge goes. Also as mentioned before and shown in figure 5.3, we cannot be certain that we detected every issue within the case. Maybe more issues existed which we could not reveal by our chosen research method. Also in our method for detecting issues, we focused on recurring issues which are easier to detect. However also unexpected issues could have existed which had to be solved outside the project but indeed were immediately solved without any problems. These issues might not be in memory of the respondents anymore or are not clearly documented. Moreover, we retrieved unexpected issues which came to the attention during the project, whereas also unexpected issues might still reveal themselves even when the project is finished. Therefore if we assume that our research method is sufficiently valid and reliable, and that the case project was planned and managed in a professional manner, then the quantity and impact of these retrieved issues seems convincing.
This reasoning compels us to conclude from our results, that it is reasonable to assume that ERP implementations despite proper preparation and management can show significant unexpected behaviour.

5.6 Added value of an aggregate complexity research approach

In the previous sections, we discussed research goals, methodology, results and conclusions of Study 4 as standing on its own. In this section, we will discuss abilities and limitations of the explicitly applied aggregate complexity research approach in this study. I.e. what have we learned from Study 4 about an aggregate complexity research approach for ERP implementations? After all the second goal of this study was to get a first indication of the usefulness of the aggregate complexity research approach for ERP implementation research and practice. This indication can contribute to answering our research question Q3 "What are the abilities and limitations of an ***aggregate*** complexity research approach for research into ERP implementation?".

As structure for our discussion we will use the abilities and limitations of an aggregate complexity research approach, as discussed in Chapter 3:

Abilities

- It is suited for research into the behaviour of social phenomena like ERP implementations.
- It does take into account the unexpected behaviour of ERP implementations.

Limitations

- This research approach is not yet explicitly used in current ERP research, and its usefulness is not yet demonstrated.
- Methods and research areas for this type of complexity research approach into ERP implementations have to be developed.
- Research is less focused on understanding a system entirely than on dealing with uncertainties.

Ability: It is suited for research into the behaviour of social phenomena like ERP implementations.
The main purpose of this part of the research into ERP implementations was to perform exploratory research explicitly with an aggregate complexity research approach. Our results support its usefulness and suitability. Moreover, they justify additional research into this topic.
During the presentation of the research results, representatives from the case organisation indicated that indeed despite proper preparation and management unexpected issues arose, not only during the present ERP project but also in other complex but yet well-managed projects. They also acknowledged that it was a valuable insight for the case organisation that, by accepting the aggregate complexity paradigm for ERP implementations, these issues cannot be prevented by yet better preparation and learning from the past (the deterministic approach). They acknowledged that proper management techniques have to be developed to deal with these issues when they arise. For the case organisation, this awareness of the existence of these unexpected issues, and that no blame could be attached to their occurrence, was valuable.
This might be seen as being merely opinions. But these opinions were expressed by several attendants during the presentation of the results to the case organisation. So, we feel these are a first indication of the usefulness of applying an aggregate complexity research approach in ERP implementation research.

Ability: It does take into account the unexpected behaviour of ERP implementations.
In Study 4 we focused on the unexpected behaviour of ERP implementations.
When studying ERP implementations by an aggregate complexity research approach, this unexpected behaviour can be revealed. Whereas, studying ERP implementations

by a deterministic complexity research approach this unexpected behaviour, if detected, would be considered as evidence of an incomplete or inappropriate model for planning the ERP implementation. In a deterministic approach, the model should be enhanced to avoid these issues in future ERP implementations. In the end, the perfect model will come into existence. However, in an aggregate complexity approach it is considered impossible to achieve a perfect model, and as unexpected issues always can occur, it is more important to focus on methods that allow handling of these unexpected issues. Thus conclusions and ideas for further research differ between a deterministic complexity research approach and an aggregate complexity research approach. Therefore an important ability of an aggregate complexity research approach in comparison with a deterministic approach is its potential for leading to new insights and research directions.

Limitation: This research approach is not yet explicitly used in current ERP research, and its usefulness is not yet demonstrated.
Of course, the results from our case study into the aggregate complexity research approach can only indicate that this approach could contribute to enhancing understanding ERP implementations in research and practice. As stated before, the results from only one, although carefully selected, case study are not enough to make strong statements about the overall usefulness of an aggregate complexity research approach. However, it shows a very reasonable indication of its usefulness.
Therefore research should explore this further by collecting evidence for the added value of performing research by an aggregate complexity research approach. Further exploration is only possible by performing more research by explicitly chosen aggregate complexity research approaches.

Limitation: Methods and research areas for this type of complexity research approach into ERP implementations have to be developed.
When designing our research method and steps for our research, which we explicitly based on an aggregate complexity research approach, we could not turn to existing ERP research. As in our structured literature review and other known research, we did not find examples how to go forth.
Therefore we applied by our best efforts research methods which we expected to serve validity and reliability for our research. Our research methods like document analysis and interview techniques are not new to ERP research. As far as we can determine, these methods and techniques served us well. Also by the results of our research, we have no reason to assume that special methods and techniques are necessary for this complexity research approach. But nevertheless, further research would be advisable into appropriate research methods and techniques for an aggregate complexity research approach. Also as we did not find research explicitly based on an aggregate complexity research approach, we could not determine which re-

search area would be most appropriate to go deeper into by aggregate complexity research. But for our research, this was not a problem.
Therefore by our research alone, we cannot confirm this limitation.

Limitation: Research is less focused on understanding a system entirely than on dealing with uncertainties.
In our case study, we aimed at retrieving unexpected behaviour. We did not try to understand why this unexpected behaviour existed. By taking an aggregate complexity research approach, we took the position that understanding of the system, in this case, the ERP implementation at the case organisation, would be impossible. One can say we adhered to a certain paradigm. The design of our research was not focused on understanding the system and also from our results we did not extract any lessons that can be used to prevent the occurrence of issues in future projects.
Therefore our research contained this limitation.

Overall conclusion
We conclude that using an aggregate complexity research approach can be a useful approach and deliver benefits for practice and research.
Although we discovered limitations of this research approach, applying an aggregate complexity research approach in ERP implementation research has the potential for leading to new insights and research directions.

CHAPTER 6: CONCLUSIONS AND RECOMMENDATIONS

6.1 Introduction

In this chapter, we will discuss conclusions which we can derive from our overall research.
We aimed at enhancing the understanding of the complexity of ERP implementations. For this, we tried to determine whether a further understanding of the construct of complexity in the field of ERP implementation will enable us to do better research in this area. As explained in Chapter 2 we aimed at enhancing the understanding of this construct by performing research into the appropriateness of the complexity approaches as partitioned by Manson.
Our main research question is:

What is the added value of explicit application of different complexity research approaches into ERP implementation?

In section 6.2 we will first briefly discuss our conclusions and results from studies 1 to 4 independent from each other and our main research question. In section 6.3 we will discuss our conclusions regarding our main research question. We will draw the conclusions for the main research question from our literature search (discussed in Chapter 3) and the applied complexity research approaches from studies 1 to 4 (discussed in Chapters 4 and 5). In section 6.4 will we discuss the methodology used for our main research question, after which we will discuss in section 6.5 the insights we gained from our main research. Finally, in section 6.6 we will suggest topics for further research based on these insights.

6.2 Conclusions for Study 1, 2, 3 and 4

Study 1: complexity impact factors on the integration process of ERP and non-ERP systems

In Study 1 we searched for factors which might influence the complexity of the integration process of ERP systems and non-ERP systems. First, we retrieved five quantitative and 21 qualitative factors from literature. Next, we sought additional factors as well as confirmation and prioritisation by a survey among experts from a European international company. These experts rated as the top five factors:

1. Number of organisations that need to be integrated
2. Number of applications.
3. Number of project owners and stakeholder groups
4. Possibility to develop custom adapters
5. Willingness of employees to share control & ownership of processes

This survey confirms that organisations should not view integration as a purely technical matter. Three out of the five most important factors are non-technical factors. We expect that the resulting list of qualitative and quantitative factors is a useful instrument for organisations to determine and value relevant factors which influence the complexity of their integration of ERP with non-ERP systems. It can be a useful tool for recognition and structured discussion of important factors which affect the complexity of integration. Usage should provide awareness of the condition of a specific factor in a particular organisation in areas like planning, control and risk management.

Study 2: Sizing ERP Implementation Projects: An Activity-Based Approach

In Study 2 we focused on the definition of logical clusters of ERP project activities. First, we performed a small survey among ERP experts which confirmed the hypothesis that ERP projects consist of a collection of clusters of activities with their focus on implementation costs and project size. Next, we extracted 405 ERP implementation project activities from published literature. After that, we designed clusters and subclusters of activities in a metaplan session with the researchers.
The resulting main clusters are:

1. Selection
2. Project configuration
3. Project management
4. Organisational and system design
5. Configuration and installation

6. Customizing
7. Infrastructure
8. Reorganization
9. System implementation
10. Training
11. Set up maintenance

Future research should validate the results by repeating it with a different and preferably larger group of experts (which we did in Study 3). Validation can also take place by checking these activities against activities retrieved from real-life projects and checking whether activities from real-life projects can be categorised according to the established clusters of activities. Further research should of course also check whether the activities that can be found in real-life project documentation occur in the list of activities from the literature search.
The validated clusters are homogeneous groups of activities which can facilitate estimation of the critical parts of an ERP implementation project.

Study 3: an expert based taxonomy of ERP implementation activities

The study intended to set a basis for better determining the size of an ERP implementation. The premise is that by being aware of standard clusters of activities for ERP implementations, estimation of costs and time could be better supported when starting an ERP implementation.
Study 3 extended the results from Study 2 by further confirming and extending the list of ERP implementation activities from literature from Study 2. Confirmation and extension were achieved by activity clustering by ERP implementation experts, and by comparison of the resulting clustering from Study 2 with Study 3.
In this study clustering of the retrieved ERP implementation activities was performed by 11 ERP implementation expert by an online card sorting technique.

The resulting clusters on the highest level retrieved by Study 3 are:

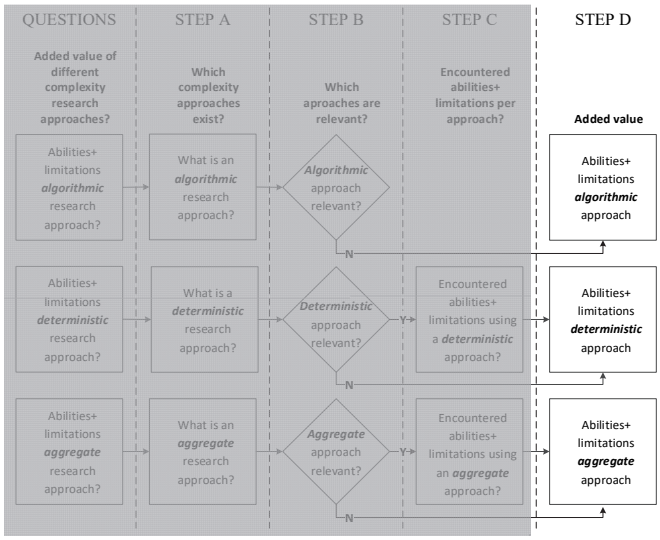
1. Software and vendor selection
2. Project initialization
3. System configuration
4. Implementation and support
5. Testing
6. Training
7. Change management
8. Technical implementation
9. Project communication
10. Business Process Design
11. Data conversion
12. Blueprinting

The detailed sorting results show two possible levels of abstraction for the clustering. We assume that the adopted level of abstraction by a participating expert is a major factor. We also assume that there is no theoretically sound grouping of ERP implementation activities. The resulting taxonomy in this research is a taxonomy solely based on expert judgment. Therefore, this taxonomy should be confirmed and enhanced by the use of empirical data from ERP implementation projects. Nevertheless as was the case with Study 2, this taxonomy shows homogeneous groups of activities which can facilitate estimation of the critical parts of an ERP implementation project.

Study 4: exploratory research into the existence of unexpected issues in ERP implementations

Our goal in this study was to perform exploratory research to discover the presence of unexpected behaviour by demonstrating unexpected issues that are clearly out-of-scope for an ERP implementation project and can only be solved outside of the project itself.
For that, we carefully selected and performed a case study within a large public body. We detected seven unexpected and significant issues as a result. We conclude from these results that it is reasonable to assume that ERP implementations show unexpected behaviour despite proper preparation and management.

6.3 Overall conclusions for our main research question



We concluded in our structured literature research that the research community, which discusses ERP implementations, clearly considers ERP implementations as complex. The statement that ERP implementations are considered complex is very common in papers covering this subject. Despite that, it is remarkable that only one paper from our structured literature research (Ghosh & Skibniewski, 2010) discussed the construct of complexity in relationship to ERP implementation. And even this paper failed to provide an in-depth discussion that could be considered helpful for our research.
Therefore we conclude:

Although the terms 'complex' and 'complexity' are very often used in ERP research results, a definition and discussion of the construct of complexity in the context of ERP implementation is insufficiently addressed.

(Chapter 3)

This conclusion encouraged us to conduct research into the construct of complexity for ERP implementation. We expect that a better understanding of ERP implementation complexity as a construct may contribute to better handling of this complexity in practice. This better understanding may also guide research into this area.
First, we considered it necessary to determine the meaning of complexity of ERP implementation. In Chapter 3, we formulated a definition for ERP implementation complexity based on a philosophical definition of complexity:

> "That property of an ERP implementation which makes it difficult to formulate its overall behaviour, even when given almost complete information about its activities, resources, (sub) products, stakeholders and their interrelations and the associated necessary organisational changes".

However, we realised that a mere definition of complexity for ERP implementations is too limited to perform explicit research into this area for the construct of complexity. Examining different complexity research approaches can provide a more comprehensive means to study ERP implementation phenomena.
Therefore we conclude:

We need a clear and practical differentiation of complexity research approaches for performing structured research into the complexity of ERP implementations.
(Chapter 3)

Hence we searched for an operationalization of complexity research approaches. We encountered Manson's differentiation of research approaches (Manson, 2001). We considered it a clear, practical and well-known operationalization. We chose to adopt Manson's differentiation as a basis for our research. The results of our case research show that this differentiation led to new insights in ERP implementation research and research methodology.
Therefore we conclude:

Manson's differentiation of complexity research approaches is clear and meaningful in the context of ERP implementation.
(Chapter 3)

Manson differentiates the following approaches:

- algorithmic complexity research approach
- deterministic complexity research approach
- aggregate complexity research approach

We compared the characteristics of Manson's complexity approaches to the characteristics of ERP implementation to be able to decide whether every complexity approach was worthwhile exploring in our research.
First, we looked at the algorithmic complexity research approach. We noticed a limited conformity between the characteristics of the algorithmic complexity approach

and ERP implementation. As a result, we decided not to explore the algorithmic complexity research approach any further.
Next, we looked at the deterministic and aggregate complexity approaches and concluded that their characteristics matched with ERP implementation for the greater part. As a result, we decided to explore these approaches further in our research.
Therefore we conclude:

From a theoretical point of view, an algorithmic complexity research approach provides limited added value when used as a principle for understanding the real complexity of ERP implementation.

From a theoretical point of view, the deterministic and aggregate complexity approaches can add value when used as a principle for understanding the real complexity of ERP implementation.

(Chapter 3)

We compared the characteristics of the deterministic and aggregate complexity research approaches with the research we analysed in our structured literature review. None of the papers showed an implicit or explicit aggregate complexity research approach for their research into ERP implementation. Also, we did not encounter an explicit deterministic complexity research approach in these papers.
Therefore we conclude:

Within mainstream ERP implementation research implicitly the deterministic complexity research approach prevails, and we presume that the aggregate complexity research approach is missing.

(Chapter 3)

We compared the characteristics of the deterministic complexity research approach with the characteristics of ERP implementation and formulated its abilities and limitations. These were validated by the results of studies 1, 2 and 3.
Therefore we conclude:

A deterministic complexity research approach can be a useful approach for research into the complexity of ERP implementations.

A deterministic complexity research approach incorporates the following abilities and limitations:

Abilities

- *This research approach prevails in current ERP research and has retrieved useful results.*
- *Methods and research areas for this type of complexity research approach into ERP implementations are well-known.*
- *A vast amount of this type of research is available as a base for future research.*

Limitations

- *It is difficult to model the behaviour of social phenomena like ERP implementations.*
- *It is difficult to take into account unexpected changes in the ERP implementation field, by which the research results may become obsolete.*
- *It can be difficult to retrieve historical data about ERP implementations required for keeping your insight up-to-date.*

(Chapter 3 and 4)

Similar to the deterministic complexity research approach, we compared the characteristics of the aggregate complexity research approach with the characteristics of ERP implementation and formulated abilities and limitations. These were validated by the results of Study 4.
Therefore we conclude:

An aggregate complexity research approach can be a useful approach for research into the complexity of ERP implementations.

An aggregate complexity research approach incorporates the following abilities and limitations:

Abilities

- *It is suited for research into the behaviour of social phenomena like ERP implementations.*
- *It does take into account the unexpected behaviour of ERP implementations.*

Limitations

- *This research approach is not yet explicitly used in current ERP research, and its usefulness is not yet demonstrated.*
- *Research is less focused on understanding a system entirely than on dealing with uncertainties.*

(Chapter 3 and 5)

Our main research question is: "What is the added value of explicit application of different complexity research approaches into ERP implementation?".
The algorithmic complexity research approach causes for ERP research an unwarranted reduction of reality. Therefore we consider an algorithmic complexity research approach of little value for ERP research.
ERP research mainly uses the deterministic complexity research approach. Application of a deterministic complexity research approach provides practice with useful guidelines to manage the complexity of ERP implementations. A deterministic approach has already proven added value for ERP implementation complexity research. However, practice and research should always bear in mind its limitations, which we made explicit by our research.
The aggregate complexity research approach is, as far as we could detect by the time of this writing not applied in current ERP implementation research, although it is suited for research into the behaviour of social phenomena like ERP implementations. As demonstrated by its abilities, the aggregate complexity research approach has added value. The aggregate complexity research approach is a new type of approach for ERP implementation research and offers a potential for leading to new insights

and research directions additional to the deterministic approach. Therefore we consider the application of the aggregate complexity research approaches next to the current deterministic approaches of high added value for ERP implementation research. Its application should lead to a better understanding of the complexity of ERP implementations.

6.4 Discussion overall methodology

In Chapter 4 and 5, we discussed the methodology, reliability and validity separately for every study, based on the level of their individual research questions and results. We justified the reliability and internal validity for our main research question by our overall research design as presented in Chapter 2. In this section, we will discuss the overall methodology of the results for our main research question.

In Chapter 3 we explored complexity itself as a construct, and explored also the complexity of ERP implementation as a construct. We encountered several definitions of complexity and chose Edward's definition. Edward's complexity definition is a clear definition based on sound research and generally applicable. We considered his definition well applicable and suitable for our research. In our literature search we were unable to retrieve a definition of the complexity of ERP implementation. Therefore, we designed a definition of complexity of ERP implementation based on Edward's complexity definition. Although we this definition was useful for us for our research, we have not further validated the usefulness of this definition in our research field.

We used Manson's differentiation as a basis for our research into complexity research approaches. Manson's differentiation is based on different complexity paradigms. A paradigm determines a person's perception of reality and a person's actions based on this perception. Therefore we intended to investigate the influence of specific complexity paradigms on ERP research. We investigated this influence by theoretically determining the abilities and limitations for every research approach, as well as by performing and evaluating research by such a specific complexity research approach. We already discussed in our conclusions that Manson's differentiation was appropriate for our research. However, we did not extensively search for the best differentiation in complexity science. Although we chose Manson's differentiation for good reasons, an alternative differentiation also could have proven to be beneficial. We are aware that Manson's differentiation determined our perspective and that perhaps other differentiations or further detailing of the three complexity approaches are possible.

Based on Manson's differentiation, we explored in detail only the deterministic and aggregate complexity research approaches. The algorithmic complexity research approach we judged, based on reasoning alone, to be not useful for further exploration in our research. Therefore the conclusions drawn on the algorithmic complexity research approach were not based on empirical research as the others were.

In Study 4 (the case study into unexpected behaviour in Chapter 5) our purpose was to perform research based on the aggregate complexity research approach. In discussing the methodology of this study, we used well-known ERP research methods.

Methods that are used in the deterministic complexity research approach as well. We also concluded that these served us well. However, in our research we did not explore which research methods would be best suitable for research performed by an aggregate complexity research approach. We are therefore unable to determine whether we performed the research in Study 4 in the best possible manner. Application of optimal methods for an aggregate complexity research approach might have resulted in higher quality of results and conclusions. Therefore, we recommend research into the best methods for the aggregate complexity research approach as a topic relevant for future research.

Based on Manson's differentiation we provided a theoretical discourse on the abilities and limitations of the deterministic and aggregate ERP research approaches. We aimed at further validating these theoretical abilities and limitations by conducting or evaluating conducted ERP research. In our literature study, we encountered research based only on the deterministic complexity research approach. Moreover, we detected within this literature no papers written with an explicit intention to show abilities and limitations of the deterministic research approach. Every researcher performs research with specific goals and a specific angle of attack. A researcher will report his findings by his adopted view and only report what is necessary to support the answer to his research question. This specificity makes it unlikely that a paper will provide sufficient information for determination of abilities and limitations for the deterministic complexity research approach. We expected that we could not extract the abilities and limitations of the deterministic complexity research approach from existing papers. Hence, we expected that by personally conducting ERP studies, we would have access to all relevant details and backgrounds of the research and could identify abilities and limitations in a more reliable manner. For that reason, we used three self-conducted ERP studies to confirm the abilities and limitations for the deterministic complexity research approach. However, we did not explicitly choose these three studies for their specific usefulness to a better understanding of abilities and limitations of the deterministic complexity research approach. We designed the three research questions for our first three studies by gaps which we detected in current ERP research. Our prime directive in our choice of the research questions for these individual studies was the usefulness of our contributions to research and practice for understanding the complexity of ERP implementation, independent of our main research goal for complexity approaches.
The findings of our three studies confirm the abilities and limitations we expected for the deterministic complexity research approach. However as already discussed in Chapter 1, the research area for ERP implementation is vast and consists of various topics. Therefore we also could have performed three other studies for which the individual study results could have proven equally useful. But as our three studies and their used deterministic complexity research approach can be regarded as matching with main ERP research, we have no reason to assume that other studies would have

revealed different abilities and limitations for the deterministic complexity research approach.
In our structured literature review, we did not encounter any research based on an aggregate complexity research approach. Although, as discussed in Chapter 3, it seemed an approach which might lead to novel insights. Therefore, we intended to confirm the abilities and limitations by also performing self-conducted research by an aggregate approach. But we could not rely on gaps in ERP research when designing a study from which we could extract conclusions for our research into the value of the aggregate complexity approach. Although, one might argue that the lack of ERP complexity research based on an aggregate complexity research approach might itself be considered a gap. Thus, we used for Study 4 a different method for determination of our research question. We sought the design of the research question in the principles of the aggregate approach itself. The underlying paradigm of the aggregate complexity research approach is based on the inevitable emergence of unexpected behaviour. Therefore based on this paradigm, we designed our research question considering whether or not we could demonstrate this unexpected behaviour for ERP implementation. In contrast, the deterministic paradigm would direct us to the conclusion that unexpected behaviour indicates that the model used for planning an ERP implementation would be incomplete or that managing the ERP implementation would have failed. In other words, the model for prediction and/or the management of the implementation failed. But in the end, by adding more studies, we would be able to entirely grasp all aspects of an ERP implementation, and be able to manage such an implementation perfectly. However, according to the aggregate paradigm, we may assume that an ERP implementation still can show unexpected behaviour despite sound planning, prediction models and management. Therefore Manson's differentiation has inspired us to the design of our research question for Study 4.
We carefully selected our case in Study 4. We have no solid basis to declare this case as being representative for the average well run implementation project. But we also have no reason to dispute this. The case might not represent all possible types of ERP implementation, but all characteristics identified did not seem out of place. The findings of Study 4 show useful results on their own and validate nearly all abilities and limitations we expected of the aggregate complexity research approach.
Study 4 clearly demonstrates the existence of unexpected behaviour in ERP implementation, and therefore the usefulness to adopt the aggregate complexity paradigm. Further studies may enhance methods for recognising such behaviours and design guidelines to deal with this kind of uncertainty.

6.5 Discussion concerning insights main research question

In this section, we will discuss the insights about our main research question, which we have gained by the results and conclusions of our literature review and self-conducted studies.

6.5.1 Limitations of past experiences, checklists and prediction methods

In Chapter 1 we discussed that practice is in need of guidelines for planning and management of ERP implementation projects. Practice gladly uses results from ERP research, often as checklists (for instance: critical success factors and risks). Furthermore, practice also likes to use evidence-based forecasting methods for costs, time and success of their ERP implementation projects. ERP implementation projects like to benefit from these checklists and methods when preparing and managing the project. However, practice also expects that if their project complies with these checklists and methods, and the project is also handled and managed in the most professional manner, the chance of success will be very high or perhaps even that compliance is a warranty for success. The (top) management of ERP implementing organisations will presume that if an ERP project is designed and managed according to the best scientifically proven guidelines, they can confidently rely on the project to handle any problems. Therefore they might assume that from their part only monitoring and budget control are necessary. However, this might be a false sense of security both on the project and management side, which is a consequence of (implicitly) using a deterministic paradigm. This paradigm encourages the expectation that someday the perfect or near-perfect model can be designed. This expectation is based on the premise that general rules can be extracted from past experiences. Therefore, in the end, it should be possible to grasp all factors and design methods which will reduce the complexity of ERP implementation to a well controllable factor.
Organisations will strive to avoid mistakes they made the last time. We expect that organisations like to believe that their mistakes caused their problems and that avoiding these mistakes in future will prevent problems. For instance, they may consider as mistakes: being insufficiently prepared or possessing too little experience for the project. As in general it is expected that one learns from their mistakes, organisations also expect that by learning they will not make these mistakes again and as a consequence will not experience the same resulting problems again. However, according to the aggregate paradigm, unexpected things are not always caused by mistakes but are simply inherent to complex systems. Unexpected things will always happen despite a proper organisation and management of a project. However, if organisations only adhere to the deterministic paradigm, then they will assume that insufficient planning and management of the project caused these problems. As a consequence, the project manager or experts will be blamed for these unexpected problems. These organisations will continue to cling to the assumption that these

mistakes, due to learning, will no longer be made in their future projects, and therefore resulting problems will never occur again.

Also in their past, organisations sometimes already frequently carried out straightforward (technically)-IT projects within their organisation. These experiences might provide an implementing organisation with a false sense of security about its abilities for implementing an ERP system. They might assume that the development and implementation of the software and instructions for the use of the software to the users are the most important activities. Of course, in general, some software development will be carried out in ERP implementation projects. Nevertheless, the most challenging and riskful activities are the inevitable changes of the organisation to be able to use the ERP system effectively. If organisations consider an ERP implementation mainly a straightforward (technically)-IT project, there is a risk that these changes will be neglected or downplayed, whereas changing an organisation is particularly the most problematic task of an ERP implementation project
For example, in our case from Study 4, the introduction of the timesheet module of their ERP system was considered a replacement of an in-use standard Excel spreadsheet. Managers and employees used their individual copy of this Excel spreadsheet to keep track of hours worked and leaves. During transferal of this in essence simple functionality to the ERP system, the project discovered that the spreadsheet was also used as an informal rewarding system for managers to their subordinates. This informal and flexible system clearly was not possible in their chosen ERP system because of authorizations and the complete integration of data with other modules. Implementing this functionality would have led to inconsistencies. As a consequence, this hidden and informal but significant business process for rewards became a critical obstacle to acceptance during the implementation of that ERP module. The project management team did not have the authorization to change or terminate this informal reward method, or any means to ensure that the new ERP system would support this reward method.

Also, an important factor that is often forgotten when relying on own experiences, checklists and methods is the fact that these experiences, lists and methods were extracted from past and different projects. ERP implementations are performed all over the world. They are carried out in different cultures and under different conditions. Still, researchers try to incorporate the common characteristics in their checklists and methods. There is a risk that some elements in these lists or prediction methods only apply within a specific context. In addition, ERP implementation already exists for over 20 years. During that period, both the world and also IT have changed significantly. As is customary in research, every scientific research relies on the results of previous research. Previous results, which might be irrelevant to present time and new technical environments. Although the concept of ERP implementation still exists after 20 years, the contents of this concept have changed. As already mentioned in this

thesis, during this period the technology of ERP systems has changed dramatically. Deployment changed from mainframe through server based to cloud based. Also, the environment in which ERP functionality is required has changed significantly. In the past, information systems supported an organisation within the boundaries of that organisation. In the present, the supply chain network economy and therefore tight integration with other organisations, require different business processes and IT capabilities. In comparison with a few years ago, factors affecting the success and other guidelines and methods for ERP implementation in practice must be inferred from different environments and experiences.

The world has changed, not only during 20 years but even during an ongoing ERP project, the world and organisation itself are constantly subject to change. Success factors, to which the project at the beginning complied, can have declined during the project. Only a few projects continuously monitor whether the terms and circumstances at the start of the project still exist during the project. Often great pressure upon projects exists to complete the project within the originally stipulated time and budget. In most cases, project managers are severely judged on these aspects. We already discussed in the aggregate complexity research approach that a complex system is emergent, has subsystems with their separate goals, learns from the past and will be influenced by its surroundings. Thus it is possible that an organisation only realises some potential benefits of its use of an ERP system during the implementation phase. Benefits which the organisation previously was unaware of, or even could not be aware of. If a project then still pursues based on the original optimal design values and goals, there is a great danger that these are no longer the optimal values and goals for the organisation as a whole.

The deterministic complexity paradigm is a useful paradigm which leads to beneficial outcomes for research and practice. Well analysed past experiences can be put to use to prevent making the same mistakes again. However, considering the items discussed above, in our opinion *managing an ERP implementation project with exclusively a deterministic complexity paradigm in mind bears a risk of a false sense of security when relying only on factors and directives derived from deterministic ERP implementation complexity research*.

6.5.2 The ERP implementation organisational roles should endorse the aggregate complexity paradigm

We have explicitly demonstrated in Study 4 that unexpected problems can arise in an ERP implementation, even though an ERP project is correctly designed and controlled. As discussed in Chapter 5 we could not expect to be able to reveal all unexpected problems in a project. By using our model we only could find the easily detectable problems, as it were “the top of the iceberg”. Nevertheless, in Study 4 seven

unexpected problems were indisputably confirmed by the organisation, which we consider an impressive number. Our case organisation was professionally designed to cope with large projects and had hired experts to compensate for missing skills. Despite that, serious unexpected problems arose that could not be resolved within and by the project. These were problems for which it was hard to identify who was responsible and who had the authority to determine and carry out the necessary solutions.

The interviewees all referred to the top management of the organisation for a decision on and execution of solutions. Analysis based on a deterministic paradigm would conclude that apparently the case project was inadequately designed for this type of problems, and therefore it also was not clear who was responsible for solving these problems. As a consequence, these problems needed to be escalated to the top. However, with the aggregate complexity paradigm in mind, a fully predictable and ideal project is an unrealistic idea. The aggregate paradigm states that, despite a presumed perfect project design and execution, unexpected problems can be expected to occur. Therefore if we reflect upon these problems through an aggregate complexity paradigm, we must conclude that the unexpected problems which appeared in the case project are normal for an adequately planned and executed ERP project.

However, even if unexpected problems are considered normal within an aggregate complexity paradigm, still it is essential for practice to have one or more general coping mechanism for handling these unexpected problems in ERP implementation. In the realm of project management, it is customary that a steering committee is responsible for making decisions on matters for which the project manager has insufficient authority, influence or knowledge. However, Study 4 also indicates that even the authority and influence of the steering committee can be insufficient to handle these unexpected issues. In our case, it was clear that these specific issues only could be handled and solved by the top management of that organisation. Only its top management could oversee all business units, weigh the impact of the exposed problems, seek possible solutions acceptable for the entire organisation, and finally approve and implement the necessary solutions. Therefore we expect that the involvement of top management is an important coping mechanism. This expectation is also supported by the fact that in ERP research on critical success factors (CSF's) "top management support" is always considered the most important CSF. Concluding by the results of our case, we would like to describe the support of top management as making a real commitment to the project by a detailed understanding of the overall implications of the project, intensively following its progress and actively make decisions on behalf of the project. This commitment is in contrast to the fire-and-forget mentality which might be adopted by the top management when considering their ERP implementation merely as a well predictable technology implementation project. An ERP implementation affects the entire organisation and, more importantly, also changes this organisation. Only top management can foresee the consequences of these changes

and can make the correct decisions to support the project for that matter. They must be prepared to take novel and perhaps risky decisions.
Thus we suggest that the role of top management in ERP implementations should rather be a coaching role than only an initiating and controlling role. They must be actively involved in such large projects, instead of settling for cheerleading. The consequences of an ERP implementation for an implementing organisation are vast and, as unexpected issues likely will occur, top management should constantly and intensively feel involved and stay involved. If the rationale for this suggestion is correct, it could clarify why top management support is indicated as the most important CSF and also further refine what is considered top management support as a CSF.

However, not only the top management should embrace the aggregate paradigm regarding ERP implementation and in accordance fulfil their role as a coach. The entire project organisation should be aware of this paradigm. If project managers or project members are not aware that unforeseen problems, which can only be solved outside the boundaries of the project or influence of the steering committee, may arise during the project, then they will keep on trying to solve these problems within their limited range of influence. Solving only within their range of influence carries a risk of improper solutions, by which other problems may arise as soon as the ERP system is in production. Alternatively and perhaps even worse, if a problem is entirely ignored; the consequences can emerge later with severe effects. Therefore, project managers, project members and members of the steering committee should be aware of this paradigm and timely engage top management without restraint and risk of censure.

In addition, the stakeholders should incorporate this paradigm in their attitude to ERP implementations. After all, the expectations of the stakeholders are shaped by the clearly defined goals and deliverables in the original project plan. In most cases, it will not please stakeholders if these goals and deliverables, because of unexpected problems, become rather different during the project. Nonetheless, stakeholders may not always perceive such deviations as negative. Changes in the project and deliverables caused by progressive insight into the capabilities of an ERP package can also create new opportunities for stakeholders. Therefore, introducing the aggregate complexity paradigm to the stakeholders from the beginning will also support their acceptance of necessary changes and maintain their confidence in the project.
In our opinion *top management, members of the ERP implementation project and other stakeholders should be aware of the paradigm that not all aspects of an ERP implementation can be planned and controlled in advance and perform their role accordingly.*

6.5.3 Explicit application of complexity research approaches

In our structured literature review, we concluded that ERP research considers ERP implementation as complex. The definition of Edmonds suggests that 'complex' im-

plies that there can be unexpected behaviour even though we seem to understand all the elements and structure of a language expression, which in our case is ERP implementation. Manson shows that research can approach complexity from various complexity paradigms. As far as we can determine, we approached ERP implementation research from a novel perspective by explicitly discussing these different approaches and determining their value for ERP implementation research.

We concluded that a research approach based only on the algorithmic paradigm is too limited for ERP implementation research. After all, this paradigm assumes a fixed or limited context. The model behind this paradigm is based on a reduced worldview. Because we were interested in actual observable behavioural phenomena, we considered the algorithmic complexity research approach not that worthwhile to be useful in our research into the complexity of ERP implementation.

The deterministic paradigm uses a broader perspective. Research by a deterministic complexity research approach is very suitable for conversion of experience into rules. These rules can increase the chances of success and predictability of future ERP implementations. We consider the deterministic paradigm as the most widely, although implicitly, used paradigm in ERP implementation complexity research. While we expect that researchers are sufficiently aware of the limitations of their results, like checklists and methods, we expect that practice when using these results is less or even unaware of these limitations. Therefore, we also advise that researchers should more prominently inform practice of the limitations of their checklists and models.

We did not detect any ERP implementation research by an aggregate complexity research approach in our structured literature search. However, we expect research according to the aggregate complexity approach to be suitable to determine how an organisation may deal with unexpected events that occur during an ERP implementation. For explaining the difference in paradigm between the deterministic and aggregate research approach, let us take as an example: “the danger that the lights go out”. Research by a deterministic complexity approach would examine: “How can we prevent the lights from going out? Though, if the lights still go out unexpectedly, what are the procedures to follow?”. The aggregate complexity research approach would examine: “Are we resilient and/or how can we be resilient enough to deal with a situation that when the lights unexpectedly go out, we did not anticipate that and also do not have standard procedures for that?”.

In summary, we consider it important that next to the deterministic paradigm; both science and practice also become aware of the aggregate complexity paradigm and explicitly apply aggregate complexity research approaches in ERP implementation research. The aggregate complexity paradigm can offer researchers the possibility to create explicit awareness of possible unexpected behaviour of a complex social system, in our case an ERP implementation. This awareness can be created without the necessity to capture reality in a model that must be suitable for any situation.

We expect it would be valuable that researchers, before performing research into ERP implementation, first explicitly opt which complexity approaches make the best sense

for their research goal. We expect it also to be valuable that they weigh the limitations and abilities against each other. This conscious choice and awareness may lead to new research directions and new solutions. For example, we would not have designed our research question for Study 4 if we had not been aware of the aggregate complexity research approach and the paradigm on which it is based. We expect that new research directions for ERP implementation will originate caused by explicit choice and application of complexity research approaches. We also expect, as was slightly illustrated by the example of the CSF "support top management", that results from deterministic research approaches can be explained or confirmed by aggregate complexity research approaches.

And finally, although in our research we explicitly linked complexity research approaches to ERP research, we expect that research outside ERP also would benefit from a conscious appliance.

In our opinion *research into the complexity of ERP implementation should be performed by explicit awareness of a deterministic or an aggregate complexity research approach*.

6.6 Recommendations for further research

As discussed in Chapter 3, we picked Manson's differentiation for practical reasons as a basis for our research. We did not systematically explore all possible and most adequate complexity research approaches. The results of our case research do not directly impel us to search for a better differentiation or enhance this differentiation. However, it is remarkable that already by using Manson's differentiation we were able to, in our opinion, gain new insights in ERP implementation research and research methodology. Therefore we believe it would also be useful to *look more profound at other theoretical differentiations for complexity approaches*. Maybe it is helpful to detail Manson's differentiation further and therefore create even more useful subdifferentiations of complexity research approaches. Alternatively, perhaps an entirely different theoretical differentiation derived from complexity research could provide us with more insights.

We explored two of Manson's research complexity approaches by evaluating and performing self-conducted research with that specific research approach in mind. In our discussion of these research approaches, we recommend that researchers make explicit choices in which specific research complexity approach they will use. In that way, they can explicitly address the abilities and limitations of each approach. Therefore we consider it advisable to *perform additional research into supporting researchers how to decide on the best research complexity approach and based on this approach select the optimal method for this complexity approach*.

In Chapter 3 we argued that we considered further exploration of the algorithmic complexity research approach for our present research not sensible. Nevertheless, it still could be useful to those who wish to study issues abstracted or idealised from real life situations in pursuit of elementary or fundamental logic constructs. *Therefore we recommend to explore further the value of the algorithmic complexity research approach* for research and practice by explicitly performing studies with this algorithmic complexity research approach in mind.

As far as we are aware, we are the first to explicitly introduce and apply an aggregate complexity research approach in an ERP research study. We concluded that application of this research approach is of value. Also, we noticed that a deterministic complexity research approach prevails in ERP research. Therefore we recommend performing *more research into ERP implementation by explicitly applying an aggregate complexity research approach.*

We performed this research specifically for ERP implementations. ERP implementation can be considered a social system. In other research areas, the behaviour of social systems also initiates research subjects. For instance research into other large projects

which have a big impact on organisations could also benefit from the differentiation into research complexity approaches. Therefore, *we expect that explicitly applying complexity paradigms in other research areas might also enhance understanding of the complexity of other research subjects.*

The default research paradigm used in ERP research is a deterministic one. Performing research by an aggregate complexity research approach might also reveal gaps in current research or explain results of current research. For instance, why do specific critical success factors exist? Alternatively, why is top management support the most important CSF? The two complexity research approaches might complement each other. Therefore, further research should be *performed by the aggregate complexity research approach explaining existing results of research by the deterministic complexity research approach*.

The paradigm behind the aggregate complexity research approach considers the ERP implementation as a system which shows unexpected behaviour. Therefore, performing research into management mechanisms which can better manage this type of behaviour seems obvious. For instance, project management and results from research, like CSFs and prediction methods, are commonly used management tools for ERP implementation. If we contemplate on control of ERP implementation with an aggregate complexity research approach in mind, we should consider that control of ERP implementation always will have to deal with the unexpected. Of course in project management also risk management and applying sufficient slack in planning and resources will meet some unexpected behaviour. However, this is limited to where a project might expect some unexpected behaviour in advance (for example, if a task takes longer than anticipated). However in ERP implementations also unexpected behaviour can exist of a more serious nature. For instance, the project scope needs to change or, an organisational unit suddenly refuses to participate in a project although the unit committed itself in advance to the project, or critical business processes unexpectedly change. Possibly other control mechanisms than the ones from project management are more suited for dealing with these unexpected issues. For instance, program management owns control mechanisms that might cope with this unexpected behaviour. As is discussed in "Gower handbook of programme management" (Reiss et al., 2006) an aspect of programs is: "Exist in a world that is constantly changing. These changes need to be constantly monitored and their impact on the programme and its projects controlled and managed". Ribbers also gave the suggestion that program management might be a suitable control mechanism for ERP implementations (Ribbers & Schoo, 2002). Therefore it seems relevant to *perform more research into the relevance of appropriate control mechanisms based on an aggregate paradigm for ERP implementation*.

In Study 4 we observed that unexpected behaviour of an ERP implementation can lead to significant changes in the organisation itself. A major commitment of top management seems obvious in these cases. We argued that top management should coach these changes. Therefore it seems relevant to perform research into whether coaching by top management supports handling of unexpected problems in ERP implementations. If coaching is relevant, it also seems worthwhile to *perform research into the mechanisms by which top management can perform a coaching role during ERP implementations*.

Not only top management needs to deal with the unexpected behaviour of ERP implementation. Therefore, an organisation as a whole should also be prepared to expect the unexpected and deal with that unexpected behaviour. In other words: organisations should be resilient to this unexpected behaviour. Therefore we expect research into the resilience of organisations against the unexpected behaviour of ERP implementation to be relevant. Currently, we already started *research into the resilience of organisations for unexpected issues during ERP implementations*.

If you want to hear the universe laugh …

tell it your plans!

REFERENCES

Ahmad, M. M., & Pinedo Cuenca, R. (2013). Critical success factors for ERP implementation in SMEs. *Robotics and Computer-Integrated Manufacturing, 29*(3), 104-111.

Ahmadi, S., Yeh, C.-H., Martin, R., & Papageorgiou, E. (2015). Optimizing ERP readiness improvements under budgetary constraints. *International Journal of Production Economics, 161*, 105-115.

Al-Mashari, M., Al-Mudimigh, A., & Zairi, M. (2003). Enterprise resource planning: A taxonomy of critical factors. *European Journal of Operational Research, 146*, 352-364.

Al-Rawashdeh, T. A., Al'azzeh, F. M., & Al-Qatawneh, S. M. (2014). Evaluation of ERP Systems Quality Model Using Analytic Hierarchy Process (AHP) Technique. *Journal of Software Engineering and Applications, 7*(4), 225-232.

Aladwani, A. M. (2001). Change management strategies for successful ERP implementation. *Business Process Management Journal, 7*(3), 266-275.

Alhadeff-Jones, M. (2008). Three Generations of Complexity Theories: Nuances and ambiguities. *Educational Philosophy and Theory, 40*, 66-82.

Alhirz, H., & Sajeev, A. S. M. (2015). Do cultural dimensions differentiate ERP acceptance? A study in the context of Saudi Arabia. *Information Technology & People, 28*(1), 163-194.

Aloini, D., Dulmin, R., & Mininno, V. (2007). Risk management in ERP project introduction: Review of the literature. *Information & Management, 44*(6), 547-567.

Aloini, D., Dulmin, R., & Mininno, V. (2012a). Modelling and assessing ERP project risks: a Petri Net approach. *European Journal of Operational Research, 220*(2), 484-495.

Aloini, D., Dulmin, R., & Mininno, V. (2012b). Risk assessment in ERP projects. *Information Systems, 37*(3), 183-199.

Alshawi, S., Themistocleous, M., & Almadani, R. (2004). Integrating diverse ERP systems: a case study. *Journal of Enterprise Information Management, 17*(6), 454 - 462.

Altamony, H., Al-Salti, Z., Gharaibeh, A., & Elyas, T. (2016). The relationship between Change Management Strategy and Successful Enterprise Resource Planning (ERP) Implementations: A Theoretical Perspective. *International Journal of Business Management and Economic Research, 7*(4), 690-703.

Amid, A., Moalagh, M., & Ravasan, A. Z. (2012). Identification and classification of ERP critical failure factors in Iranian industries. *Information Systems, 37*(3), 227-237.

Arb, R. v. (1997). *Vorgehensweisen und Erfahrungen bei der Einführung von Enterprise-Management-Systemen dargestellt am Beispiel von SAP R/3.* (PhD), Institute für Wirtschaftsinformatik der Universität Bern, Bern.

Arif, M., Kulonda, D., Jones, J., & Proctor, M. (2005). Enterprise information systems: technology first or process first? *Business Process Management Journal, 11*(1), 5-21.

Baccarini, D. (1996). The concept of project complexity--a review. *International Journal of Project Management, 14*, 201-204.

Barki, H., & Pinsonneault, A. (2005). A Model of Organizational Integration, Implementation Effort, and Performance. *Organization Science, 16*(2), 165-179.

Basoglu, N., Daim, T., & Kerimoglu, O. (2007). Organizational adoption of enterprise resource planning systems: A conceptual framework. *Journal of High Technology Management Research, 18*(1), 73-97.

Basu, R., Upadhyay, P., & Dan, P. k. (2011a). Factors influencing ERP implementation in Indian SMEs: An empirical analysis. *Management Science Letters, 1*(2), 89-98.

Basu, R., Upadhyay, P., & Dan, P. K. (2011b). Identification of Factors Affecting Successful ERP Implementation in Indian SMEs. *International Journal of Information, 3*, 94-101.

Berchet, C., & Habchi, G. (2005). The implementation and deployment of an ERP system: An industrial case study. *56*(6), 588-605.

Bhatt, G. D. (2000). An empirical examination of the effects of information systems integration on business process improvement. *International Journal of Operations & Production Management, 20*(11), 1331 - 1359.

Blickstein, I., Drezner, J. A., Libicki, M. C., McInnis, B., McKernan, M., Nemfakos, C., . . . Wong, C. (2012). Root Cause Analyses of Nunn-McCurdy Breaches *RAND Corporation monograph series* (Vol. 2, pp. 116). Santa Monica: RAND National Defense Research Institute.

Bollou, F., Balogun, E., & Usang, I. (2012). *Eradicating complexity in software interface for increased productivity: Increasing Effectiveness of Enterprise Systems*. Paper presented at the Proceedings of the International Conference on Software Engineering Research and Practice (SERP).

Bolstorff, P. A. (2002). Supply Chain: A Framework for Expanding the Human Resource Development Professional's Role in Technology Implementations. *Advances in Developing Human Resources, 4*(4), 533-549.

Bose, I., Pal, R., & Ye, A. (2008). ERP and SCM systems integration: the case of a valve manufacturer in China. *Information & management, 45*(4), 233-241.

Botta-Genoulaz, V., Millet, P. A., & Grabot, B. (2005). A survey on the recent research literature on ERP systems. *Computers in Industry, 56*(6), 510-522.

Boudreau, M.-c., Robey, D., Marie-Claude, B., & Daniel, R. (1999). Organizational transition to enterprise resource planning systems: theoretical choices for process research. *Proceedings of the 20th international conference on Information Systems*, 291-299.

Bradford, M., & Florin, J. (2003). Examining the role of innovation diffusion factors on the implementation success of enterprise resource planning systems. *International journal of accounting information systems, 4*, 205-225.

Brödner, P. (2009). The misery of digital organisations and the semiotic nature of IT. *AI & SOCIETY, 23*(3), 331-351.

Bruges, P. (2002). ERP Implementation Methodologies. *MSIS, 488*.

Bueno, S., & Salmeron, J. L. (2008). TAM-based success modeling in ERP. *Interacting with Computers, 20*(6), 515-523.

Buonanno, G., Faverio, P., Pigni, F., & Ravarini, A. (2005). Factors affecting ERP system adoption: A comparative analysis between SMEs and large companies. *Journal of Enterprise Information Management, 18*(4), 384-426.

Burg van der, J. G., Vos, J., Schimmel, R., & Poecke van, A. A. H. (2013). Eindrapportage Programma SPEER Terugblik op de invoering van ERP bij Defensie (pp. 55). Den Haag: Ministerie van Defensie.

Cakular, & Wijngaarden, v. (2002). Succesvolle EAI *Informatie* (Vol. November 2002, pp. 72-76).

Capra, F. (1996). *The Web of Life: A New Scientific Understanding of Living System*. New York: Anchor Books.

Champy, J., & Weger, J. (2005). Reengineering: the second time around. *Strategy & Leadership, 33*(5), 53-56.

Chang, J. Y., Jiang, J. J., Klein, G., & Wang, E. T. (2014). Do too many goals impede a program? A case study of enterprise system implementation with multiple interdependent projects. *Information & management, 51*(4), 465-478.

Clegg, B., & Wan, Y. (2013). Managing enterprises and ERP systems: a contingency model for the enterprization of operations. *International journal of operations & production management, 33*(11/12), 1458-1489.

Cotran, K., Buchmeiser, U., Seguin, J.-L., & Pelster, B. (2005). HR's role in implementing JTI's global ERP system: How HR's focus on change management was key to a successful ERP project. *Strategic HR Review, 4*(5), 24-27.

Coyne, I. (1997). Sampling in Qualitative Research. Purposeful and Theoretical Sampling; Merging or Clear Boundaries? *Journal of Advanced Nursing, 26*(3), 623-630

Daneva, M. (2010). Balancing uncertainty of context in ERP project estimation: an approach and a case study. *Journal of Software Maintenance and Evolution: Research and Practice, 22*, 329-357.

Davenport, T. H. (1998). Putting the enterprise into the enterprise system. *Harvard business review, 76*(4).

Davenport, T. H., & Brooks, J. D. (2004). Enterprise systems and the supply chain. *Journal of Enterprise Information Management, 17*(1), 8-19.

Davidson, D. J. (2010). The Applicability of the Concept of Resilience to Social Systems: Some Sources of Optimism and Nagging Doubts. *Society & Natural Resources, 23*(12), 1135-1149.

Dezdar, S., & Ainin, S. (2011). Examining ERP implementation success from a project environment perspective. *Business Process Management Journal, 17*(6), 919-939.

Dhillon, G. S., Caldeira, M., & Wenger, M. R. (2011). Intentionality and power interplay in IS implementation: the case of an asset management firm. *The journal of strategic information systems, 20*(4), 438-448.

Dittrich, Y., Vaucouleur, S., & Giff, S. (2009). ERP Customization as Software Engineering: Knowledge Sharing and Cooperation. *IEEE Software, 26*(6), 41-47.

Donnadieu, G., Durand, D., Neel, D., Nunez, E., & Saint-Paul, L. (2003). L'Approche systémique: de quoi s' agit-il. *Union Europeenne de Systemique*, 11.

Edmonds, B. (1999). *Syntactic Measures of Complexity.* (PhD), University of Manchester, Manchester.

Ehie, I. C., & Madsen, M. (2005). Identifying critical issues in enterprise resource planning (ERP) implementation. *Computers in Industry, 56*, 545-557.

Eijnatten, v. F. (2003). *Chaordic systems thinking: chaos and complexity to explain human performance management*. Paper presented at the First International Conference on Performance Measures, Benchmarking and Best Practices in the New Economy, “Business Excellence ’03”, Portugal.

El Amrani, R., Rowe, F., & Geffroy-Maronnat, B. (2006). The effects of enterprise resource planning implementation strategy on cross-functionality. *Information Systems Journal, 16*(1), 79-104.

Esteves, J., Pastor, J., & Casanovas, J. (2002). *Monitoring Business Process Redesign in ERP Implementation Projects*. Paper presented at the Eighth Americas Conference on Information Systems.

Esteves, J., & Pastor, J. A. (2001). Analysis of critical success factors relevance along SAP implementation phases. *Seventh Americas Conference on Information Systems*.

Ettlie, J. E., Perotti, V. J., Joseph, D. A., & Cotteleer, M. J. (2005). Strategic predictors of successful enterprise system deployment. *International journal of operations & production management, 25*(10), 953-972.

Fadlalla, A., & Amani, F. (2015). A keyword-based organizing framework for ERP intellectual contributions. *Journal of Enterprise Information Management, 28*(5), 637-657.

Fan, Y. S., Shi, W., & Wu, C. (1999). Enterprise wide application integration platform for CIMS implementation. *Journal of Intelligent Manufacturing, 10*(6), 587-601.

Fontana, R. M., & Neto, A. I. (2009). ERP systems implementation in complex organizations. *JISTEM - Journal of Information Systems and Technology Management (Online), 6*(1), 61-92.

Francalanci, C., Iar, C. H., & Lanci, A. F. (2001). Predicting the implementation effort of ERP projects: empirical evidence on SAP/R3. *Journal of Information Technology, 16*, 33-48.

Françoise, O., Bourgault, M., & Pellerin, R. (2009). ERP implementation through critical success factors' management. *Business Process Management Journal, 15*(3), 371-394.

Frimpon, M. F. (2012). A Project Approach to Enterprise Resource Planning Implementation. *International Journal of Business and Management, 7*(10), 116.

Fui-Hoon Nah, F., Lee-Shang Lau, J., & Kuang, J. (2001). Critical factors for successful implementation of enterprise systems. *Business Process Management Journal, 7*(3), 285 - 296

Gadakh, V. S., & Mawale, M. B. (2011). Enterprise Resource Planning (ERP) of Canteen using USA Principle: A case study. *International Journal of Applied Engineering Research, Dindigul, 2*(1), 183-189.

Garg, P., & Agarwal, D. (2014). Critical success factors for ERP implementation in a Fortis hospital: an empirical investigation. *Journal of Enterprise Information Management, 27*(4), 402-423.

Ghosh, S., & Skibniewski, M. J. (2010). Enterprise Resource Planning Systems Implementation as a Complex Project: A Conceptual Framework. *Journal of Business Economics and Management*, 533-549.

Gibbert, M., & Ruigrok, W. (2010). The ‘‘What’’ and ‘‘How’’ of Case Study Rigor: Three Strategies Based on Published Work. *Organizational Research Methods, 13*(4), 710-737.

Gibson, N., Holland, C. P., & Light, B. (1999). *Enterprise Resource Planning: A Business Approach to Systems Development.* Paper presented at the Proceedings of the 32nd Hawaii International Conference on System Sciences - 1999.

Gilfillan, M., & Fowler, A. (2003). A Framework for Stakeholder Integration in Higher Education Information Systems Projects. *Technology Analysis & Strategic Management, 15*(4), 468-489.

Goossenaerts, J. B. M., Zegers, A. T. M., & Smits, J. M. (2009). A multi-level model-driven regime for value-added tax compliance in ERP systems. *Computers in Industry, 60*(9), 709-727.

Grabski, S. V., & Leech, S. A. (2007). Complementary controls and ERP implementation success. *International journal of accounting information systems, 8*(1), 17-39.

Grabski, S. V., Leech, S. A., & Schmidt, P. J. (2011). A Review of ERP Research: A Future Agenda for Accounting Information Systems. *Journal of Information Systems, 25*(1), 37-78.

Griend, v. d. P., & Kusters, R. J. (2012). Integration test effort in SAP R/3 systems. *Journal of Software: Evolution and Process, 24*(4), 421-435.

Gulledge, T. (2006). What is integration? *Industrial Management & Data Systems, 106*(1), 5 - 20.

Gustafsson, T., & Ollila, M. (2003). Expert Consultation in the Preparation of a National Technology Programme. *Systems Analysis Laboratory, Helsinki University of Technology*, 31.

Hallikainen, P., Kimpimäki, H., & Kivijärvi, H. (2006). Supporting the Module Sequencing Decision in the ERP Implementation Process. *Proceedings of the 39th Hawaii International Conference on System Sciences - 2006*, 1-10.

Hansen, T. (2006). *Multidimensional Effort Prediction for ERP System Implementation.* Paper presented at the On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops.

Haynes, R. (2006). *Design Knowledge as a Learning Resource.* Paper presented at the First International Conference on Design Science Research in Information Systems and Technology.

Helo, P., Anussornnitisarn, P., & Phusavat, K. (2008). Expectation and reality in ERP implementation: consultant and solution provider perspective. *Industrial Management & Data Systems, 108*(8), 1045-1059.

Hendricks, K. B., Singhal, V. R., & Stratman, J. K. (2006). The impact of enterprise systems on corporate performance: A study of ERP, SCM, and CRM system implementations. *Journal of Operations Management, In Press, Corrected Proof*.

Hertogh, M., & Westerveld, E. (2009). Playing with Complexity: managemen and organisation of large infrastructure projects (Vol. Phd). Rotterdam: Erasmus Universiteit Rotterdam.

Holland, C. R., & Light, B. (1999). A critical success factors model for ERP implementation. *Software, IEEE, 16*(3), 30-36.

Holsapple, C. W., Wang, Y.-M., & Wu, J.-H. (2005). Empirically Testing User Characteristics and Fitness Factors in Enterprise Resource Planning Success. *International Journal of Human-Computer Interaction, 19*(3), 325-342.

Howard, M. S. (1994). *Quality of Group Decision Support Systems: a comparison between GDSS and traditional group approaches for decision tasks.* Eindhoven University of Technology, Eindhoven.

Hsu, S.-C., Wang, T.-C., Chang, T.-H., & Chang, J. (2009). Application of incomplete linguistic preference relations in predicting the success of ERP implementation. *2009 IEEE International Conference on Fuzzy Systems*: Ieee.

Huang, Z. (2010). A compilation research of ERP implementation critical success factors. *Issues in Information Systems, 11*(1), 507-512.

Huin, S. F. (2004). Managing deployment of ERP systems in SMEs using multi-agents. *International Journal of Project Management, 22*(6), 511-517.

Hwang, D. W., Min, H., & Chong, A. (2015). Identifying the drivers of enterprise resource planning and assessing its impacts on supply chain performances. *Industrial Management & Data Systems, 115*(3), 541-569.

Hwang, Y., & Leitch, R. A. (2005). Balanced Scorecard: Evening the Odds of Successful BPR. *IT Professional, 7*, 24-30.

Jääskelainen, K., & Pau, L.-f. (2009). ERP project's internal stakeholder network and how it influences the project's outcome. *Social Science Electronic Publishing*.

Jacobs, J., Moll, J. v., & Kusters AC (Aarnout); Trienekens, J. J., R J (Rob); Brombacher. (2007). *Effects of Virtual Product Development on Product Quality and their Influencing Factors.* (Phd), Technische Universiteit Eindhoven, Eindhoven.

Janssens, G., Hoeijenbos, M., & Kusters, R. (2011). *Complexity impact factors on the integration process of ERP and non ERP systems: A basis for an evaluation instrument*. Paper presented at the ICSOFT 2011 - Proceedings of the 6th International Conference on Software and Data Technologies, Seville, Spain.

Janssens, G., Kusters, R., & Heemstra, F. (2008a). Sizing ERP Implementation Projects: An Activity-Based Approach. *International Journal of Enterprise Information Systems, 4*, 23.

Janssens, G., Kusters, R., & Heemstra, F. (2008b). A small survey into the importance of and into a concept for estimating effort-related costs of ERP implementation projects. *Working papers Management Sciences*, 14.

Janssens, G., Kusters, R., & Heemstra, F. (2010). Specifying general activity clusters for ERP projects aimed at effort prediction. In A. Gunasekaran & T. Shea (Eds.), *Organizational Advancements through Enterprise Information Systems: Emerging Applications and Developments* (pp. 57-79). New York: Business Science Reference.

Kanchymalay, K., Krishnan, R., Arif, F., Amiruddin, S., Salam, S., & Hashim, U. R. (2013). The Extent of ERP Customization towards User Satisfaction in Daily Operation for Manufacturing Companies. *Journal of Computers, 8*(7), 1788-1792.

Kanungo, S., & Bagchi, S. (2000). Understanding User Participation and Involvement in ERP Use. *Journal of Management Research, 1*(1), 47.

Karimi, J., Somers, T. M., & Bhattacherjee, A. (2007). The Impact of ERP Implementation on Business Process Outcomes: A Factor-Based Study. *Journal of Management Information Systems, 24*(1), 101-134.

Kasvi, J. J. J., Vartiainen, M., Pulkkis, A., & Nieminen, M. (2000). The role of information support systems in the joint optimization of work systems. *Human Factors and Ergonomics in Manufacturing & Service Industries, 10*, 193-221.

Khalil, T., Dominic, P., & Fadzil bin Hassan, M. (2010). *Decision support system framework for implementation of Enterprise Resource Planning (ERP) system.*

Paper presented at the Information Technology (ITSim), 2010 International Symposium in.

Kien, S. S., & Lian, Y. P. (2009). Building Enterprise Integration Through Enterprise Resource Planning Systems. *ICIS 2009 Proceedings*, 18.

Klaus, H., Rosemann, M., & Gable, G. (2000). What is ERP? *Information Systems Frontiers, 2*, 141-162.

Klesse, M., Wortmann, F., & Schelp, J. (2005). Success factors for application integration. *Wirtschaftsinformatik, 47*(4), 259-267.

Koch, C. (2001). BPR and ERP: Realising a vision of process with IT. *Business Process Management Journal, 7*, 258.

Kovács, G. L., & Paganelli, P. (2003). A planning and management infrastructure for large, complex, distributed projects—beyond ERP and SCM. *Computers in Industry, 51*(2), 165-183.

Kræmmergaard, P., & Rose, J. (2002). Managerial Competences for ERP Journeys. *Information Systems Frontiers, 4*(2), 199-211.

Krumbholz, M., & Maiden, N. (2001). The implementation of enterprise resource planning packages in different organisational and national cultures. *Information Systems, 26*, 185-204.

Kumar, V., Kumar, U., & Maheshwari, B. (2002). ERP systems implementation: best practices in Canadian government organizations. *Government Information Quarterly, 19*(2), 147-172.

Kumar, V., Maheshwari, B., & Kumar, U. (2003). An investigation of critical management issues in ERP implementation: empirical evidence from Canadian organizations. *Technovation, 23*(10), 793-807.

Kusters, R., & Heemstra, F. (2007). Determining the costs of ERP implementation. *CONCEPT*, 9.

Kusters, R. J., Heemstra, F. J., & Jonker, A. (2009). ERP Implementation Costs: A Preliminary Investigation *Enterprise Information Systems* (Vol. Volume 12, pp. 95-107): Springer Berlin Heidelberg.

Kwahk, K.-Y. (2006). ERP Acceptance: Organizational Change Perspective *Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06) Track 8* (pp. 10).

Kwon, O. B., & Lee, J. J. (2001). A multi-agent intelligent system for efficient ERP maintenance. *Expert Systems with Applications, 21*(4), 191-202.

Kwon, T. H., & Zmud, R. W. (1987). Unifying the fragmented models of information systems implementation *Critical issues in information systems research* (pp. 227-251): John Wiley \& Sons, Inc.

Lam, W. (2005). Investigating success factors in enterprise application integration: a case-driven analysis. *European Journal of Information Systems, 14*(2), 175-187.

Larsen, T. J. (2009). A multilevel explanation of end-user computing satisfaction with an enterprise resource planning system within an international manufacturing organization. *Computers in Industry, 60*(9), 657-668.

Latvanen, H., & Ruusunen, R. (2001). Management of Risks in an ERP Implementation Project (pp. 20): Turku School of Economics.

Lee, J. C., & Myers, M. D. (2004). Dominant actors, political agendas, and strategic shifts over time: a critical ethnography of an enterprise systems implementation. *Journal of Strategic Information Systems, 13*(4), 355-374.

Lee, T. (2003). *Complexity Theory in Axiomatic Design.* (PHD), Massachusetts institue of technology.

Lemahieu, W., Snoeck, M., & Michiels, C. (2003). An enterprise layer-based approach to application service integration. *Business Process Management Journal, 9*(6), 760 - 775.

Lengnick-Hall, C. A., Lengnick-Hall, M. L., & Abdinnour-Helm, S. (2004). The role of social and intellectual capital in achieving competitive advantage through enterprise resource planning (ERP) systems. *Journal of Engineering and Technology Management, 21*(4), 307-330.

Li, Y., Liao, X. W., & Lei, H. Z. (2006). A knowledge management system for ERP implementation. *Systems Research and Behavioral Science, 23*(2), 157-168.

Light, B., Holland, C. P., & Wills, K. (2001). ERP and best of breed: a comparative analysis. *Business Process Management Journal, 7*(3), 216-224.

Little, A., & Best, P. J. (2003). A framework for separation of duties in an SAP R/3 environment. *Managerial Auditing Journal, 18*(5), 419-430.

Lungu, I., & Vătuiu, T. (2007). Computer assisted audit techniques. *Annals of the University of Petrosani Economics*, 217-224.

Lyytinen, K., Newman, M., & Al-Muharfi, A.-R. A. (2009). Institutionalizing enterprise resource planning in the Saudi steel industry: a punctuated socio-technical analysis. *Journal of Information Technology, 24*(4), 286-304.

M. Beheshti, H., K. Blaylock, B., A. Henderson, D., & G. Lollar, J. (2014). Selection and critical success factors in successful ERP implementation. *Competitiveness review, 24*(4), 357-375.

Mabert, V. A., Soni, A., & Venkataramanan, M. A. (2003). Enterprise resource planning: Managing the implementation process. *European Journal of Operational Research, 146*(2), 302-314.

Mabert, V. A., Soni, A., & Venkataramanan, M. A. (2005). Model based interpretation of survey data: A case study of enterprise resource planning implementations. *In Press, Corrected Proof*.

Magee, C., & de Weck, O. (2004). *Complex system classification*. Paper presented at the Fourteenth Annual International Symposium of the International Council On Systems Engineering (INCOSE).

Magnusson, J., Nilsson, A., & Carlsson, F. (2004). Forecasting ERP Implementation Success - Towards a Grounded Framework *ECIS 2004 Proceedings* (pp. 9).

Maguire, S., Ojiako, U., & Said, A. (2010). ERP implementation in Omantel: a case study. *Industrial Management & Data Systems, 110*(1), 78-92.

Maheshwari, B., Kumar, V., & Kumar, U. (2010). Delineating the ERP institutionalization process: go-live to effectiveness. *Business Process Management Journal, 16*(4), 744-771.

Manduca, C. A., & Mogk, D. W. (2006). *Earth and mind: How geologists think and learn about the earth* (Vol. 413): Geological Society of America.

Manson, S. M. (2001). Simplifying complexity: a review of complexity theory. *Geoforum, 32*, 405-414.

Markus, M. L., & Tanis, C. (2000). Chapter 10: The Enterprise System Experience—From Adoption to Success. *Framing the Domains of IT Management: Projecting the Future Through the Past, Pinnaflex Educational Resources, Inc.*, 173-207.

Markus, M. L., & Tanis, C. (2003). The Enterprise System Experience - From Adoption to Success. *Pinnaflex Educational Resources*, 173-207.

Markus, M. L., Tanis, C., & van Fenema, P. C. (2000). Enterprise resource planning: multisite ERP implementations. *Communications of the ACM, 43*(4), 42-46.

Marler, J. H., & Liang, X. (2012). Information technology change, work complexity and service jobs: a contingent perspective. *New Technology, Work and Employment, 27*(2), 133-146.

Marnewick, C., & Labuschagne, L. (2005). A conceptual model for enterprise resource planning (ERP). *Information Management & Computer Security, 13*(2), 144-155.

Martin, R., Mauterer, H., & Gemunden, H. G. (2002). Classifying the benefits of ERP systems in the manufacturing industry. *Wirschaftsinformatik, 44*(2), 109-116.

Masbahi, M. E., Bahsani, S., & Semma, A. (2012). Enterprise Resource Planning (ERP) and Integration of Purchase, Inventory and Sale processes: Proposition of a Common Minimal Model. *International Journal of Computer Science Issues (IJCSI), 9*(5), 181-190.

Masini, A., & Wassenhove, L. N. (2009). ERP competence-building mechanisms: an exploratory investigation of configurations of ERP adopters in the European and US manufacturing sectors. *Manufacturing & service operations management, 11*(2), 274-298.

McAdam, R., & Galloway, A. (2005). Enterprise resource planning and organisational innovation: a management perspective. *Industrial Management & Data Systems, 105*(3), 280-290.

McCarthy, I. P., Rakotobe-Joel, T., & Frizelle, G. (2000). Complex systems theory: applications and promises for manufacturing organizations. *International Journal of Manufacturing Technology and Management, 2*, 559-579.

Meredith, J. R., & Mantel Jr, S. J. (2011). *Project management: a managerial approach* (Eighth ed.): John Wiley & Sons.

Miltenburg, J. (2001). Computational complexity of algorithms for MRP and JIT production planning problems in enterprise resource planning systems. *Production Planning & Control, 12*(2), 198-209.

Ming-Chang Lee, J.-F. C. J.-F. C. (2011). An Entropy Decision Model for Selection of Enterprise Resource Planning System. *International Journal of Computer Trends and Technology, 1*, 162-169.

Møller, C. (2005). ERP II: a conceptual framework for next-generation enterprise systems? *Journal of Enterprise Information Management, 18*(4), 483-497.

Møller, C., Kræmmergaard, P., & Rikhardsson, P. (2004). A Comprehensive ERP bibliography - 2000-2004. *Department of Marketing, Informatics and Statistics, Aarhus School of Business, IFI Working paper series*(12), 54.

Moon, Y. (2007). Enterprise Resource Planning (ERP): a review of the literature. *Int. J. Management and Enterprise Development, 4*, pp.235-264.

Motwani, J., Subramanian, R., & Gopalakrishna, P. (2005). Critical factors for successful ERP implementation: Exploratory findings from four case studies. *56*, 529-544.

Nandhakumar, J., Rossi, M., & Talvinen, J. (2005). The dynamics of contextual forces of ERP implementation. *Journal of Strategic Information Systems, 14*(2), 221-242.

Nazemi, E., Tarokh, M., & Djavanshir, G. (2012). ERP: a literature survey. *The International Journal of Advanced Manufacturing Technology, 61*, 999-1018.

Neto, A. I. (1996). *Contribuition a la gestion des systèmes complexes.* (PhD), Université D'Aix-Marseille III.

Newell, S., Huang, J. C., Galliers, R. D., & Pan, S. L. (2003). Implementing enterprise resource planning and knowledge management systems in tandem: fostering efficiency and innovation complementarity. *Information and Organization, 13*(1), 25-52.

Ngai, E. W. T., Law, C. C. H., & Wat, F. K. T. (2008). Examining the critical success factors in the adoption of enterprise resource planning. *Computers in Industry, 59*(6), 548-564.

Ojiako, U., Papadopoulos, T., Thumborisuthi, C., & Fan Yang, Y. (2012). Perception variability for categorised risk factors. *Industrial Management & Data Systems, 112*(4), 600-618.

Özen, M. (2011). *Product Structure Modeling in an Enterprise Resource Planning System.* (MSc), Université D'Aix-Marseille III.

Palanisamy, R. (2007). Organizational culture and knowledge management in ERP implementation: An empirical study. *JOURNAL OF COMPUTER INFORMATION SYSTEMS, 48*(2), 100-120.

Pan, M.-J., & Jang, W.-Y. (2008). Determinants of the adoption of enterprise resource planning within the technology-organization-environment framework: Taiwan's communications industry. *The Journal of Computer Information Systems, 48*(3), 94.

Parr, A., & Shanks, G. (2000). A model of ERP project implementation. *Journal of Information Technology, 15*(4), 289-303.

Parrish, J. L. (2008). *Sensemaking in information systems: toward a sensemaking inquiring system.* (PhD), University of Central Florida.

Paul, C. L. (2008). A modified delphi approach to a new card sorting methodology. *Journal of Usability Studies, 4*(1), 24.

Poba-Nzaou, P., Raymond, L., & Fabi, B. (2008). Adoption and risk of ERP systems in manufacturing SMEs: a positivist case study. *Business Process Management Journal, 14*(4), 530-550.

Rajagopal, P. (2002). An innovation--diffusion view of implementation of enterprise resource planning (ERP) systems and development of a research model. *Information & Management, 40*(2), 87-114.

Ram, J., Corkindale, D., & Wu, M. L. (2013). Enterprise Resource Planning Adoption: Structural Equation modeling analysis of antecdants. *JOURNAL OF COMPUTER INFORMATION SYSTEMS, 54*(1), 53-65.

Ranganathan, C., & Brown, C. V. (2006). ERP Investments and the Market Value of Firms: Toward an Understanding of Influential ERP Project Variables. *Information Systems Research, 17*(2), 145-161.

Reiss, G., Anthony, M., Chapman, J., Leigh, G., Rayner, P., & Pyne, A. (2006). *Gower Handbook of Programme Management*: Gower Publishing, Ltd.

Remington, K., & Pollack, J. (2007). *Tools for complex projects*: Gower Publishing, Ltd.

Ribbers, P. M. A., & Schoo, K.-C. (2002). Program management and complexity of ERP implementations. *Engineering Management Journal, 14*, 45-52.

Robertson, J. (2001). Information design using card sorting. *Intranet Design Series*, 12.

Robey, D., Ross, J. W., & Boudreau, M.-C. (2002). Learning to Implement Enterprise Systems: An Exploratory Study of the Dialectics of Change. *Journal of Management Information Systems, 19*(1), p17, 30p.

Rogers, E. M. (2010). *Diffusion of innovations*: Simon and Schuster.

Rose, J., & Schlichter, B. R. (2013). Decoupling, re-engaging: managing trust relationships in implementation projects. *Information Systems Journal, 23*(1), 5-33.

Ruivo, P., Oliveira, T., & Neto, M. (2014). Examine ERP post-implementation stages of use and value: empirical evidence from Portuguese SMEs. *International journal of accounting information systems, 15*(2), 166-184.

Salmeron, J. L., & Lopez, C. (2010). A multicriteria approach for risks assessment in ERP maintenance. *The Journal of Systems & Software, 83*(10), 1941-1953.

Salmeron, J. L., & Lopez, C. (2012). Forecasting Risk Impact on ERP Maintenance with Augmented Fuzzy Cognitive Maps. *IEEE Transactions on Software Engineering, 38*(2), 439-452.

Sammon, D., & Adam, F. (2005). Towards a model of organisational prerequisites for enterprise-wide systems integration: Examining ERP and data warehousing. *Journal of Enterprise Information Management, 18*(4), 458-470.

Sarker, S., & Lee, A. S. (2003). Using a case study to test the role of three key social enablers in ERP implementation. *Information & Management, 40*(8), 813-829.

Schlichter, B. R., & Kraemmergaard, P. (2010). A comprehensive literature review of the ERP research field over a decade. *Journal of Enterprise Information Management, 2*, 486-520.

Schniederjans, D. (2013). Successful ERP implementation: an integrative model. *Business Process Management Journal, 19*, 364-398.

Scott, J. E. (1999). The FoxMeyer Drugs' Bankruptcy: Was it a Failure of ERP? *Americas Conference on Information Systems, August 13-15, Milwaukee,*, 223-225.

Shafiei-Monfared, S., & Jenab, K. (2012). Fuzzy Complexity Model for Enterprise Maintenance Projects. *IEEE Transactions on Engineering Management, 59*(2), 293-298.

Sharif, A. M., Elliman, T., Love, P. E. D., & Badii, A. (2004). Integrating the IS with the enterprise: key EAI research challenges. *Journal of Enterprise Information Management, 17*(2), Journal of Enterprise Information Management.

Sharif, A. M., Irani, Z., & Love, P. E. D. (2005). Integrating ERP using EAI: a model for post hoc evaluation. *European Journal of Information Systems, 14*(2), 162-174.

Shaul, L., & Tauber, D. (2013). Critical Success Factors in Enterprise Resource Planning Systems: Review of the Last Decade. *ACM Computing Surveys, 45*(4), 55-55:39.

Soliman, F., & Youssef, M. (2001). The impact of some recent developments in e-business on the management of next generation manufacturing. *International journal of operations & production management, 21*(5/6), 538-564.

Somers, T. M., & Nelson, K. G. (2001). Proceedings of the Annual Hawaii International Conference on System Sciences The Impact of Critical Success Factors across the Stages of Enterprise Resource Planning Implementations (CONF34 ed.).

Somers, T. M., & Nelson, K. G. (2004). A taxonomy of players and activities across the ERP project life cycle. *Information & management, 41*, 257-278.

Somers, T. M., & Ragowsky, A. (2003). The impact of system complexity and implementations characteristics on top IT managements' s perceptions of enterpise system success. *Wayne State University, School of Business Administration, Department of Information Systems and Manufacturing*.

Sommer, R. A. (2003). Business process flexibility: a driver for outsourcing. *Industrial Management & Data Systems, 103*(3), 177-183.

Spiteri, K. J., Luca, C. L., Reynolds, T., & Wilson, G. (2012). Defining a baseline complexity model for ERP systems over SaaS. *Journal of Internet Technology and Secured Transactions*, 86-92.

Standish, R. K. (2008). Concept and Definition of Complexity. *arXiv preprint arXiv:0805.0685, 805*, 1-12.

Štemberger, M. I., Vukšić, V. B., & Kovačič, A. (2009). Business Process Modelling as a Critical Success Factor in Implementing an ERP System. *South East European Journal of Economics and Business, 4*(2), 89-96.

Stensrud, E. (2001). Alternative approaches to effort prediction of ERP projects. *Information and Software Technology, 43*(7), 413-423.

Strong, D. M., & Volkoff, O. (2004). A roadmap for enterprise system implementation. *Computer, 37*(6), 22-29.

Sullivan, L., & Porter, R. (2006). Implementing student information systems. *New Directions for Higher Education, 2006*(136), 35-51.

Sumner, M. (2000). Risk factors in enterprise-wide/ERP projects. *Journal of Information Technology, 15*(4).

Taskin, N. (2011). *Flexibility and strategic alignment of enterprise resource planning systems with business strategies: an empirical study.* (PhD), The University of British Columbia, Okanagan.

Tchokogué, A., Bareil, C., & Duguay, C. R. (2005). Key lessons from the implementation of an ERP at Pratt & Whitney Canada. *International Journal of Production Economics, In Press, Corrected Proof*.

Tejeida-Padilla, R., Badillo-Pina, I., & Morales-Matamoros, O. (2010). A Systems Science Approach to Enterprise Resources Planning Systems. *Systems Research and Behavioral Science, 27*(1), 87-95.

Teltumbde, A. (2000). A framework for evaluating ERP projects. *International Journal of Production Research, 38*(17), 4507-4520.

Themistocleous, M., & Irani, Z. (2002). Novel taxonomy for application integration. *Benchmarking: An International Journal, 9*(2), 154 - 165.

Themistocleous, M., Irani, Z., & O´Keefe, R. M. (2001a). Benchmarking the benefits and barriers of application integration. *Benchmarking: An International Journal, 8*(4), 317 - 331.

Themistocleous, M., Irani, Z., & O´Keefe, R. M. (2001b). ERP and application integration: Exploratory survey. *Business Process Management Journal, Vol. 7*(3), 195-204.

Tomkos, T. (1999). *Formal specification and modeling of complex systems.* (PhD), Universität Hamburg, Hamburg.

Trochim, W. (1989). Concept Mapping: Soft Science or Hard Art? *Evaluation and Program Planning, 12*(1), 87-110.

Tullis, T., & Wood, L. (2004). How Many Users Are Enough for a Card-Sorting Study? *Proceeding UPA'2004, Minneapolis, June 7-11*.

Umble, E. J., Haft, R. R., & Umble, M. M. (2003). Enterprise resource planning: Implementation procedures and critical success factors. *European Journal of Operational Research, 146*(2), 241.

Verville, J., & Halingten, A. (2002). An investigation of the decision process for selecting an ERP software: the case of ESC. *Management Decision, 40*(3), 206-216.

Viskoviü, D., & Varga, M. (2008). *Bad practices in complex IT projects.* Paper presented at the Proceedings of the ITI 2008 30th Int. Conf. on Information Technology Interfaces, Cavtat.

Vosburg, J., & Kumar, A. (2001). Managing dirty data in organizations using ERP: lessons from a case study. *Industrial Management & Data Systems, 101*(1), 21-31.

Wagner, W., & Antonucci, Y. L. (2004). An analysis of the imagine PA public sector ERP project. *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on*, 8.

Wang, W., Hsieh, J., & Butler, J. E. (2006). System emergent use: a theoretical model and empirical exploration.

Ward, J., Hemingway, C., & Daniel, E. (2005). A framework for addressing the organisational issues of enterprise systems implementation. *Journal of Strategic Information Systems, 14*(2), 97-119.

Wei, C.-C., Chien, C.-F., & Wang, M.-J. J. (2005). An AHP-based approach to ERP system selection. *International Journal of Production Economics, 96*, 47-62.

Wei, C.-C., & Wang, M.-J. J. (2004). A comprehensive framework for selecting an ERP system. *International Journal of Project Management, 22*(2), 161-169.

Weston, F. C. W. J. (2001). ERP implementation and project management. *Production and Inventory Management Journal, 42*(3/4), 75.

Williams, J., Williams, M. D., & Morgan, A. (2013). A teleological process theory for managing ERP implementations. *Journal of Enterprise Information Management, 26*(3), 235-249.

Williams, T. M. (1999). The need for new paradigms for complex projects. *International Journal of Project Management, 17*(5), 269-273.

Willis, T. H., Willis-Brown, A. H., & McMillan, A. (2001). Cost containment strategies for ERP system implementations. *Production and Inventory Management Journal, 42*(2), 36.

Wong, A., Scarbrough, H., Chau, P. Y. K., & Davison, R. (2005). *Critical Failure Factors in ERP Implementation.* Paper presented at the Pacis, Bangkok, Thailand.

Wood, T., & Caldas, M. P. (2001). Reductionism and complex thinking during ERP implementations. *Business Process Management Journal, 7*(5), 387-393.

Worley, J. H., Chatha, K. A., Weston, R. H., Aguirre, O., & Grabot, B. (2005). Implementation and optimisation of ERP systems: A better integration of processes, roles, knowledge and user competencies. *Computers in Industry, 56*(6), 620-638.

Wu, F., Li, H. Z., Chu, L. K., Sculli, D., & Gao, K. (2009). An approach to the valuation and decision of ERP investment projects based on real options. *Annals of operations research, 168*(1), 181-203.

Wu, J.-H., & Wang, Y.-M. (2007). Measuring ERP success: The key-users' viewpoint of the ERP to produce a viable IS in the organization. *Computers in Human Behavior, 23*(3), 1582-1596.

Wu, W.-W., Lee, Y.-T., & Lan, L. W. (2013). On the Decision Structures and Knowledge Discovery for ANP Modeling. *International Journal of Intelligence Science, 3*, 15-23.

Xu, L., Wang, C., & Luo, X. (2006). Integrating Knowledge Management and ERP in Enterprise Information Systems. *Systems Research and Behavioral Science, 23*(2), 147.

Xue, Y., Liang, H., Boulton, W. R., & Snyder, C. A. (2005). ERP implementation failures in China: Case studies with implications for ERP vendors. *International Journal of Production Economics, 97*(3), 279-295.

Yahia, E., Aubry, A., & Panetto, H. (2012). Formal measures for semantic interoperability assessment in cooperative enterprise information systems. *Computers in Industry, 63*(5), 443-457.

Yeh, J. Y., & OuYang, Y.-c. (2010). How an organization changes in ERP implementation: a Taiwan semiconductor case study. *Business Process Management Journal, 16*(2), 209-225.

Yusuf, Y., Gunasekaran, A., & Abthorpe, M. S. (2004). Enterprise information systems project implementation: A case study of ERP in Rolls-Royce. *International Journal of Production Economics, 87*(3), 251-266.

DANKWOORD

Toen ik startte met mijn promotieonderzoek wist ik eigenlijk totaal niet waar ik aan begon!
Ik zag het vooral als het volgende project. Een project dat ik wel met een goede planning en voldoende inspanning, binnen overzienbare tijd, gestructureerd tot een goed einde zou brengen. Ik was immers bedreven in het plannen en beheersen van mijn werk en bijbehorende projecten. Ik dacht dat de voortgang daarom vooral afhankelijk zou zijn van de inspanning die ik erin zou stoppen. Echter in mijn geval, en wellicht ook in andere promotietrajecten, bleek niets minder waar te zijn.
Ik had niet kunnen bevroeden dat toen ik startte met promoveren, dit een dergelijk langdurig en natuurlijk (ja waarde lezer, deze open deur is helaas onvermijdelijk) zeer 'complex' traject zou worden. Achteraf begrijp ik dat wel. Ik heb het gevoel dat ik alle wetenschappelijke obstakels die men tijdens een promotietraject kan tegenkomen wel geraakt heb. Uiteindelijk ligt er een proefschrift van ongeveer 300 pagina's. Maar ik vermoed dat als ik alles wat ik geschreven heb (en moest weggooien) gedurende het promotietraject zou laten drukken, dit de omvang van een aantal proefschriften zou hebben.
Ik heb veel verschillende wegen tijdens het traject bewandeld. Ik moest helaas enkele keren terugkeren van doodlopende of onbevredigende weggetjes en daarna weer nieuwe ideeën uitwerken. Het oorspronkelijk idee van mijn promotoren was om een soort omvang-maat voor ERP implementaties te vinden. Daarmee zouden een heleboel aspecten van een dergelijk ERP implementatietraject voorspeld kunnen worden. Nadat ik wat dieper in de onderzoeksliteratuur gedoken was, bleek dat dit uiteindelijk toch geen reële vraag was. Via wat omzwervingen, belandde ik tenslotte in een meer wetenschapsfilosofische richting over complexiteit. Dit gebied en het nadenken hierover fascineerde mij. Het bracht mij ook veel persoonlijke inzichten. Maar ik denk dat ik het mij, en ook mijn promotoren, veel gemakkelijker had gemaakt, als ik binnen de gebaande wegen van ERP onderzoek was gebleven. Gelukkig staat daar tegenover dat het nadenken over complexiteit mij ook persoonlijk heeft geraakt en daardoor mij misschien wel het belangrijkste voordeel van het promoveren heeft gebracht. Het maakte mij er diepgaand van bewust dat niet alles maakbaar en/of beheersbaar is. Om hier mee om te gaan, zul je daarom als mens moeten accepteren dat er onvoorziene zaken gebeuren en dat je er alleen maar naar kunt streven dat je er op dat moment naar beste kunnen mee probeert om te gaan. Ook tijdens mijn promotietraject zijn er natuurlijk veel onvoorziene zaken gebeurd, zowel op persoonlijk als ook zakelijk

vlak. Door de houding aan te nemen dat dit bij de complexiteit van het leven hoort, maakte het dit voor mij gemakkelijker om hiermee om te gaan.

Hoewel de verdieping in de complexiteitstheorie volgens mij tot een zinvol en naar ik hoop ook tot nieuwe inzichten aanzettend proefschrift heeft geleid, was deze verdieping voor mij ook een nadeel. Complexiteit is geen onderwerp dat in mijn vakgebied (ERP en projectmanagement) uitgebreid behandeld wordt. Daardoor was er ook veel tijd nodig om het onderwerp te laten rijpen en de voordelen hiervan in mijn onderzoek op te nemen. Dit gold niet alleen voor mij, maar ook voor mijn promotoren die hierin met mij mee moesten groeien. Tijdens het promotietraject hebben wij samen heel wat uurtjes gediscussieerd over wat dit inzicht in complexiteit zou kunnen betekenen. Ook hebben wij veel gediscussieerd over hoe dit nu het beste in mijn onderzoek zou kunnen worden geïntegreerd. Dat waren geen eenvoudige discussies en ik was dan ook blij dat ik deze discussies altijd met mijn memorecorder opgenomen heb. Zo kon ik ieder nuance nog eens kon naluisteren en objectief bekijken. Het was moeilijk om mijn onderzoek steeds op twee abstractieniveaus te blijven beschouwen. Dat wil zeggen, aan de ene kant rechttoe rechtaan zinvol onderzoek dat in de praktijk gebruikt kan worden om ERP implementaties te verbeteren, en aan de andere kant het denken over de wijze waarop de verschillende complexiteitsparadigma's gebruikt kunnen worden in ERP onderzoek. Toen ik de complexiteitstheorie weg insloeg, had ik niet kunnen denken dat dit mijn onderzoek zo gecompliceerd zou gaan maken. Gezien de inspanning die ik voor het promotieonderzoek moest plegen en de stress die ik hierdoor ervoer, heb ik vaak zelfs getwijfeld of ik wel het promotieonderzoek zou moeten afronden. Maar ik heb nooit getwijfeld aan de zinvolheid van het bekijken van ERP implementaties middels de complexiteitstheorie of überhaupt toepassing van complexiteits-denken. Hoewel de tijd het zal moeten leren, ben ik ervan overtuigd dat ik met mijn onderzoek een deur geopend heb naar een nieuwe wijze van denken over ERP implementatie onderzoek en ERP implementaties in de praktijk.

Het was erg plezierig dat ik tijdens het promotietraject mijn inzichten al verschillende keren via presentaties aan organisaties heb kunnen meedelen. Uit de positieve reacties bleek ook dat ik iets zinvols op het spoor was en dus ook dat mijn resultaten iets voor de praktijk kunnen betekenen. Aangezien ik van mening ben dat wetenschap er vooral is om de praktijk te ondersteunen, ben ik daar heel blij mee en ga na mijn promotie zeker door met het overbrengen van mijn inzichten naar de praktijk.

Een dankwoord hoort te noemen welke belangrijke personen hierin vooral ondersteund hebben. Helaas het is niet mogelijk om iedere ondersteuning specifiek aan mijn promotietraject te relateren. Maar diegenen waar dat wel voor mogelijk is, wil ik natuurlijk op deze plaats heel graag bedanken.

Uiteraard op de eerste plaats dank aan Rob Kusters en Harry Martin die mij bij dit zeer lange en complexe traject begeleid hebben in hun rol als promotor en copromotor. Het zal niet gemakkelijk geweest zijn om samen met mij deze uitstap te maken naar het begrip complexiteit in een diepere betekenis. Ook voor hen was dat buiten hun comfortzone en betekende te komen tot nieuwe inzichten. Zij moesten op grond van mijn resultaten steeds weer inzichten bijstellen en samen nieuwe wegen zoeken. Ik wil Rob Kusters als promotor hartelijk danken voor begeleiding die hij gedurende al die jaren gegeven heeft en vooral ook dat hij in het proces is blijven geloven en blijven begeleiden. Ik heb veel geleerd van de scherpte en structuur die hij in de discussies en onderwerpen bracht en hoop dit ook voor mijn master studenten te kunnen doen en mogelijk ook als ik een keer een copromotor rol zou vervullen. Hij heeft steeds zeer standvastig de kwaliteit en diepte die hij voor dit onderwerp voor ogen had bewaakt. Harry Martin wil ik voor zijn rol als copromotor heel hartelijk danken voor het mede ondersteunen en vooral ook het meedenken in de discussies over complexiteit in de laatste periode, toen mijn onderzoek gericht was op complexiteit. Harry gaf gelukkig voldoende tegenwicht aan Rob, als de perfectie die Rob voor het onderzoek in gedachten had soms de mogelijkheid om het promotieonderzoek ook een keer praktisch af te ronden in de weg stond. Ik heb van beiden geleerd om op grond van inzichten die je krijgt naar aanleiding van onderzoek dat je gedaan hebt, dingen weg te gooien en overnieuw te beginnen. Ook al levert dit veel frustratie op en heeft zwaarwegende consequenties voor het onderzoek. Ik wil wel vermelden dat ik dat in toekomstig onderzoek niet al te vaak en in die mate meer wil meemaken ;-).
Ik ben zeker ook dank verschuldigd aan het informele promotiegroepje van onze faculteit. Kees Gelderman (en in het verleden ook Boudewijn Janssen) heeft dit uit mededogen (maar ik vermoed vooral medelijden) opgericht om de medewerkers van de faculteit die bezig zijn met een promotie te ondersteunen. Kees wil ik heel erg danken voor dit initiatief en de wijze waarop hij het functioneren van dit groepje vorm gaf. Ik ga de bijeenkomsten zeker missen. In dit groepje kon ik altijd mijn zorgen kwijt, kon ik gerustgesteld worden dat anderen soms dezelfde zorgen hadden, kon ik stoom afblazen, werden sommige zaken in een ander daglicht geplaatst en kreeg ik vaak zinvolle tips. Ik wil dan ook Martine Coun en Mimi Crijns als “mede” medewerker-promovendi hartelijk danken voor hun steun en adviezen. Ik wens beiden een snelle en vooral stress-vrije afronding van hun eigen promotietraject toe. Ik heb zeer genoten van de humor en zelfspot in ons promotiegroepje.
Ik wil ook twee van mijn afstudeerders bedanken die door hun master thesis de basis hebben gelegd voor twee in dit proefschrift opgenomen artikelen over deelonderzoeken 1 en 3, respectievelijk René Hoeijenbos en Linda van der Velde-van Moorst. Zonder de hoge kwaliteit van hun afstudeeronderzoek en resultaten waren deze onderdelen niet tot stand kunnen komen. Hoewel ik beloofd heb dat ik de resultaten van het case-onderzoek vertrouwelijk blijven, wil ik toch zeker de case organisatie en contactpersonen die dit hebben mogelijk gemaakt hartelijk danken voor de mogelijkheid om mijn onderzoek voor deelonderzoek 4 uit te voeren. Ik heb altijd alle medewer-

king gekregen en toegang tot de mensen en gegevens, waarvoor mijn grote dank. Ik zie het als het belangrijkste en persoonlijk voor mij het leukste onderdeel van het promotieonderzoek. Ik hoop dat ik via dit proefschrift ook iets aan die organisatie terug kan geven.

Ik heb mijn promotieonderzoek op en naast mijn werk en gezinsleven gedaan. In het begin had dit nog niet zo'n invloed op mijn gezinsleven. Maar ik heb gemerkt dat toen ik uiteindelijk in de complexiteitstheorie de juiste basis voor mijn onderzoek had gevonden, dit van mijn kant meer energie ging kosten en ik ook op het privévlak hiervoor behoorlijke offers moest gaan brengen.
Zoals ik ervaren heb en ook van anderen gehoord heb, is promoveren een vrij eenzame activiteit. Echter als er een thuisfront is dat stabiel is en begrip hiervoor heeft, dan is die eenzaamheid wel te verdragen. Ik wil daarom mijn kinderen Fréderique en Stef bedanken dat ze er zijn en dat ze zijn wie ze zijn. Ik ben er erg trots op dat ze als paranimfen bij de verdediging naast mij zullen staan.
En "last but not least" mag natuurlijk dank aan mijn levenspartner niet ontbreken. Jacqueline je hebt nooit geklaagd over de invloed die het promotietraject heeft gehad op de tijd, aandacht en energie van mij voor onze relatie. Achteraf ben ik daar zeer verbaasd over, want een dergelijk traject heeft altijd invloed. Als promovendus ga je immers door een aantal diepe dalen. Je hebt het promoveren altijd geaccepteerd als iets dat gewoon moest gebeuren en mij als vanzelfsprekend daar ondersteund waar dat mogelijk was. Je hebt mij nu weer voor de volle 100% terug. Ik ben blij dat onze relatie in ieder geval niet complex is. Dank je dat je dit voor mij mogelijk hebt gemaakt, maar vooral dat ik je mijn levenspartner mag noemen en samen met jou de complexiteit van het leven mag ervaren en ervan kan genieten!

CURRICULUM VITAE

Guy Janssens earned his master's degree in Industrial Engineering and Management at the Eindhoven University of Technology in 1986. After his studies, he held several positions in business. Since 2001, he has been an assistant professor at the faculty of Management, Science & Technology at the Open Universiteit. He designs courses and teaches the subjects Enterprise Resource Planning and Project Management. For these subjects he supervises students in their theses and conducts research.

APPENDICES

Appendices Chapter 3

Appendix 3.1: Performed searches into complexity and ERP implementations

Search settings and type of documents
Scholar settings for searches:

- exclusive patents
- exclusive citations

Library Open Universiteit settings for searches:

- Only with full text online
- Only scientific and peer-reviewed materials

Types of documents:

- conference report
- magazine paper
- journal paper
- NO newspaper papers
- NO book reviews
- NO book chapters
- NO professional journals

Search strings and hits
Search 1
Scholar: "Enterprise Resource Planning" AND ("complex" OR "complexity")
HITS 52,100

Search 2
Library Open Universiteit: (TitleCombined:("Enterprise Resource Planning")) AND ((TitleCombined:(complex)) OR (TitleCombined:(complexity)))
HITS 2

Search 3
Library Open Universiteit: (Abstract:(Enterprise Resource Planning)) AND ((Abstract:(complex)) OR (Abstract:(complexity)))
HITS 274
After evaluation, 73 papers selected for further investigation.

Search 4
Library Open Universiteit: ((Abstract:("Enterprise Resource Planning")) NOT (Abstract:("complex")) NOT (Abstract:("complexity"))) AND ((Fulltext:("define complex")) OR (Fulltext:("define complexity")) OR (Fulltext:("definition of complex")) OR

(Fulltext:("definition of complexity")) OR (Fulltext:("what is complex")) OR (Fulltext:("what is complexity")) OR (Fulltext:("complexity theory")) OR (Fulltext:("complex project")))
HITS 26

Search 5
Scholar: (intitle:"Enterprise Resource Planning" OR abstract:"Enterprise Resource Planning") AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory" OR "complex project)
HITS 11
After evaluation, 9 papers selected for further investigation.

Search 6
Scholar: "Enterprise Resource Planning" AND (abstract:"complex" OR abstract:"complexity")
HITS 54
After evaluation, 7 papers selected for further investigation.

Authors that define complexity according to Hertogh and Westerveld (Hertogh & Westerveld, 2009):

- Robert Axelrod
- Michael D. Cohen
- Peter Coveney
- Roger Highfield
- Robert L. Flood
- Ewart R. Carson
- Murray Gell-Mann
- Joel Moses
- Charles Perrow
- Eberhardt Rechtin
- Mark Maier
- Peter Senge
- Ralph Stacey
- John Sterman
- Joseph Sussman
- Geer Teisman
- Edward O. Wilson

Search 7
Scholar: "Enterprise Resource Planning" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 949

Search 8
Scholar: "Enterprise Resource Planning" AND "Manson" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 11
After evaluation, 0 papers selected for further investigation.

Search 9
Scholar: "Enterprise Resource Planning" AND "Edmonds" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 18
After evaluation, 1 paper selected for further investigation.

Search 10
Scholar: "Enterprise Resource Planning" AND "Robert Axelrod" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 0

Search 11
Scholar: "Enterprise Resource Planning" AND "Michael D. Cohen" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 0

Search 12
Scholar: "Enterprise Resource Planning" AND "Peter Coveney" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 0

Search 13
Scholar: "Enterprise Resource Planning" AND "Roger Highfield" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 0

Search 14
Scholar: "Enterprise Resource Planning" AND "Ewart R. Carson" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 0

Search 15
Scholar: "Enterprise Resource Planning" AND "Robert L. Flood" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 0

Search 16
Scholar: "Enterprise Resource Planning" AND "Murray Gell-Mann" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 2
After evaluation, 0 papers selected for further investigation.

Search 17
Scholar: "Enterprise Resource Planning" AND "Joel Moses" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 2
After evaluation, 0 papers selected for further investigation

Search 18
Scholar: "Enterprise Resource Planning" AND "Charles Perrow" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 10
After evaluation, 0 papers selected for further investigation

Search 19
Scholar: "Enterprise Resource Planning" AND "Eberhardt Rechtin" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 1
After evaluation, 0 papers selected for further investigation

Search 20
Scholar: "Enterprise Resource Planning" AND "Mark Maier" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 2
After evaluation, 0 papers selected for further investigation

Search 21
Scholar: "Enterprise Resource Planning" AND "Peter Senge" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 15
After evaluation, 0 papers selected for further investigation

Search 22
Scholar: "Enterprise Resource Planning" AND "Ralph Stacey" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 4
After evaluation, 0 papers selected for further investigation

Search 23
Scholar: "Enterprise Resource Planning" AND "John Sterman" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 0

Search 24
Scholar: "Enterprise Resource Planning" AND "Joseph Sussman" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 1
After evaluation, 0 papers selected for further investigation

Search 25
Scholar: "Enterprise Resource Planning" AND "Geert Teisman" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 0

Search 26
Scholar: "Enterprise Resource Planning" AND "Edward O. Wilson" AND ("define complex" OR "define complexity" OR "definition of complex" OR "definition of complexity" OR "what is complex" OR "what is complexity" OR "complexity theory")
HITS 2
After evaluation, 0 papers selected for further investigation

All searches were performed in February 2015

Appendix 3.2: Used databases for searches into complexity and ERP implementations

- Academic Search Elite (EBSCO)
- ACM Digital Library
- ACS Publications
- Business Source Premier (EBSCO)
- Cambridge University press
- Catalogue UM
- DOAJ - Directory of Open Access Journals
- EBSCO Host
- E-Journals (EBSCO)
- Emerald [management plus]
- ERIC (EBSCO)
- Google Scholar / Google Wetenschap
- GreenFILE (EBSCO)
- HeinOnline
- IEEE Digital Library
- JSTOR Business, Biological, Mathematics & Statistics Collection
- Kluwer Navigator
- Lecture Notes in Computer Science
- Legal Intelligence
- Library, Information Science & Technology Abstracts - LISTA (EBSCO)
- NARCIS - the Gateway to Dutch Scientific Information
- Nature : international weekly journal of science
- OpMaat Premium
- Overheid.nl
- Oxford Journals
- PiCarta (NCC + OLC)
- PsycArticles (EBSCO)
- Psychology and Behavioral Sciences Collection (EBSCO)
- PsycINFO (EBSCO)
- PubMed
- Regional Business News (EBSCO)
- SAGE Journals Online
- Science
- ScienceDirect (Elsevier)
- SpringerLink
- Taylor & Francis Group
- Web of Science
- Wiley Online Library

From these databases the next databases were completely covered by the library search facilities from the Open Universiteit:

- ACM Digital Library
- Cambridge University press
- DOAJ - Directory of Open Access Journals
- Emerald [management plus]
- HeinOnline
- IEEE Digital Library
- JSTOR Business, Biological, Mathematics & Statistics Collection
- Lecture Notes in Computer Science
- Nature : international weekly journal of science
- Oxford Journals
- PubMed
- SAGE Journals Online
- Science
- ScienceDirect (Elsevier)
- SpringerLink
- Taylor & Francis Group
- Wiley Online Library

Also http://dspace.ou.nl/ was completely covered.

The content of the EBSCO databases and Google Scholar were only covered by the library search facilities for full text availability:

- Academic Search Elite (EBSCO)
- Business Source Premier (EBSCO)
- EBSCO Host
- E-Journals (EBSCO)
- ERIC (EBSCO)
- GreenFILE (EBSCO)
- Library, Information Science & Technology Abstracts - LISTA (EBSCO)
- PsycArticles (EBSCO)
- Psychology and Behavioral Sciences Collection (EBSCO)
- PsycINFO (EBSCO)
- Regional Business News (EBSCO)
- Google scholar

Appendix 3.3: Retrieved papers about complexity and ERP implementations

Paper ID	Search	Paper	Full reference
1	3	(Ahmadi, Yeh, Martin, & Papageorgiou, 2015)	Ahmadi, S., Yeh, C.-H., Martin, R., & Papageorgiou, E. (2015). Optimizing ERP readiness improvements under budgetary constraints. *International Journal of Production Economics, 161*, 105-115. doi: 10.1016/j.ijpe.2014.11.020
2	3	(Aladwani, 2001)	Aladwani, A. M. (2001). Change management strategies for successful ERP implementation. *Business Process Management Journal, 7*(3), 266-275. doi: 10.1108/14637150110392764
3	4	(Alhirz & Sajeev, 2015)	Alhirz, H., & Sajeev, A. S. M. (2015). Do cultural dimensions differentiate ERP acceptance? A study in the context of Saudi Arabia. *Information Technology & People, 28*(1), 163-194. doi: 10.1108/ITP-07-2013-0127
4	3	(Davide Aloini et al., 2012b)	Aloini, D., Dulmin, R., & Mininno, V. (2012). Modelling and assessing ERP project risks: a Petri Net approach. *European Journal of Operational Research, 220*(2), 484-495. doi: 10.1016/j.ejor.2012.01.062
5	3	(Davide Aloini, Dulmin, & Mininno, 2012a)	Aloini, D., Dulmin, R., & Mininno, V. (2012). Risk assessment in ERP projects. *Information Systems, 37*(3), 183-199. doi: 10.1016/j.is.2011.10.001
6	3	(Al-Rawashdeh, Al'azzeh, & Al-Qatawneh, 2014)	Al-Rawashdeh, T. A., Al'azzeh, F. M., & Al-Qatawneh, S. M. (2014). Evaluation of ERP Systems Quality Model Using Analytic Hierarchy Process (AHP) Technique. *Journal of Software Engineering and Applications, 07*(04), 225-232. doi: 10.4236/jsea.2014.74024
7	3	(Amid et al., 2012)	Amid, A., Moalagh, M., & Ravasan, A. Z. (2012). Identification and classification of ERP critical failure factors in Iranian industries. *Information Systems, 37*(3), 227-237. doi: 10.1016/j.is.2011.10.010
8	3	(Arif, Kulonda, Jones, & Proctor, 2005)	Arif, M., Kulonda, D., Jones, J., & Proctor, M. (2005). Enterprise information systems: technology first or process first? *Business Process*

Paper ID	Search	Paper	Full reference
			Management Journal, 11(1), 5-21. doi: 10.1108/14637150510578692
9	3	(Barki & Pinsonneault, 2005)	Barki, H., & Pinsonneault, A. (2005). A Model of Organizational Integration, Implementation Effort, and Performance. *Organization Science, 16*(2), 165-179. doi: 10.1287/orsc.1050.0118
10	4	(Basoglu, Daim, & Kerimoglu, 2007)	Basoglu, N., Daim, T., & Kerimoglu, O. (2007). Organizational adoption of enterprise resource planning systems: A conceptual framework. *Journal of High Technology Management Research, 18*(1), 73-97. doi: 10.1016/j.hitech.2007.03.005
11	3	(Basu, Upadhyay, & Dan, 2011b)	Basu, R., Upadhyay, P., & Dan, P. k. (2011). Factors influencing ERP implementation in Indian SMEs: An empirical analysis. *Management Science Letters, 1*(2), 89-98. doi: 10.5267/j.msl.2011.01.003
12	3	(Basu, Upadhyay, & Dan, 2011a)	Basu, R., Upadhyay, P., & Dan, P. K. (2011). Identification of Factors Affecting Successful ERP Implementation in Indian SMEs. *International Journal of Information, 3*, 94-101.
13	5	(Blickstein et al., 2012)	Blickstein, I., Drezner, J. A., Libicki, M. C., McInnis, B., McKernan, M., Nemfakos, C., . . . Wong, C. (2012). Root Cause Analyses of Nunn-McCurdy Breaches. Volume 2: Excalibur Artillery Projectile and the Navy Enterprise Resource Planning Program, with an Approach to Analyzing Program Complexity and Risk: DTIC Document.
14	5	(Bollou et al., 2012)	Bollou, F., Balogun, E., & Usang, I. (2012). ERADICATING COMPLEXITY IN SOFTWARE INTERFACE FOR INCREASED PRODUCTIVITY.
15	4	(Bolstorff, 2002)	Bolstorff, P. A. (2002). Supply Chain: A Framework for Expanding the Human Resource Development Professional's Role in Technology Implementations. *Advances in Developing Human Resources, 4*(4), 533-549. doi: 10.1177/152342202237527
16	3	(Bose, Pal, & Ye, 2008)	Bose, I., Pal, R., & Ye, A. (2008). ERP and SCM systems integration: the case of a valve manufacturer in China. *Information & management,*

Paper ID	Search	Paper	Full reference
			45(4), 233-241. doi: 10.1016/j.im.2008.02.006
17	3	(Bradford & Florin, 2003)	Bradford, M., & Florin, J. (2003). Examining the role of innovation diffusion factors on the implementation success of enterprise resource planning systems. *International journal of accounting information systems, 4*(3), 205-225. doi: 10.1016/S1467-0895(03)00026-5
18	4	(Brödner, 2009)	Brödner, P. (2009). The misery of digital organisations and the semiotic nature of IT. *AI & SOCIETY, 23*(3), 331-351. doi: 10.1007/s00146-006-0066-1
19	3	(Bueno & Salmeron, 2008)	Bueno, S., & Salmeron, J. L. (2008). TAM-based success modeling in ERP. *Interacting with Computers, 20*(6), 515-523. doi: 10.1016/j.intcom.2008.08.003
20	3	(Buonanno, Faverio, Pigni, & Ravarini, 2005)	Buonanno, G., Faverio, P., Pigni, F., & Ravarini, A. (2005). Factors affecting ERP system adoption: A comparative analysis between SMEs and large companies. *Journal of Enterprise Information Management, 18*(4), 384-426. doi: 10.1108/17410390510609572
21	4	(Champy & Weger, 2005)	Champy, J., & Weger, J. (2005). Reengineering: the second time around. *Strategy & Leadership, 33*(5), 53-56. doi: 10.1108/10878570510616898
22	6	(Chang, Jiang, Klein, & Wang, 2014)	Chang, J. Y., Jiang, J. J., Klein, G., & Wang, E. T. (2014). Do too many goals impede a program? A case study of enterprise system implementation with multiple interdependent projects. *Information & management, 51*(4), 465-478.
23	3	(Clegg & Wan, 2013)	Clegg, B., & Wan, Y. (2013). Managing enterprises and ERP systems: a contingency model for the enterprization of operations. *International journal of operations & production management, 33*(11/12), 1458-1489. doi: 10.1108/IJOPM-07-2010-0201
24	4	(Cotran, Buchmeiser, Seguin, & Pelster, 2005)	Cotran, K., Buchmeiser, U., Seguin, J.-L., & Bill-Pelster. (2005). HR's role in implementing JTI's global ERP system: How HR's focus on change management was key to a successful ERP project. *Strategic HR Review, 4*(5), 24-27. doi:

Paper ID	Search	Paper	Full reference
			10.1108/14754390580000818
25	4	(Thomas H. Davenport & Brooks, 2004)	Davenport, T. H., & Brooks, J. D. (2004). Enterprise systems and the supply chain. *Journal of Enterprise Information Management, 17*(1), 8-19. doi: 10.1108/09576050410510917
26	3	(Dezdar & Ainin, 2011)	Dezdar, S., & Ainin, S. (2011). Examining ERP implementation success from a project environment perspective. *Business Process Management Journal, 17*(6), 919-939. doi: 10.1108/14637151111182693
27	3	(Dhillon, Caldeira, & Wenger, 2011)	Dhillon, G. S., Caldeira, M., & Wenger, M. R. (2011). Intentionality and power interplay in IS implementation: the case of an asset management firm. *The journal of strategic information systems, 20*(4), 438-448. doi: 10.1016/j.jsis.2011.09.003
28	3	(Dittrich, Vaucouleur, & Giff, 2009)	Dittrich, Y., Vaucouleur, S., & Giff, S. (2009). ERP Customization as Software Engineering: Knowledge Sharing and Cooperation. *IEEE Software, 26*(6), 41-47. doi: 10.1109/MS.2009.173
29	4	(El Amrani, Rowe, & Geffroy-Maronnat, 2006)	El Amrani, R., Rowe, F., & Geffroy-Maronnat, B. (2006). The effects of enterprise resource planning implementation strategy on cross-functionality. *Information Systems Journal, 16*(1), 79-104. doi: 10.1111/j.1365-2575.2006.00206.x
30	3	(Ettlie, Perotti, Joseph, & Cotteleer, 2005)	Ettlie, J. E., Perotti, V. J., Joseph, D. A., & Cotteleer, M. J. (2005). Strategic predictors of successful enterprise system deployment. *International journal of operations & production management, 25*(10), 953-972. doi: 10.1108/01443570510619473
31	3	(Fontana & Neto, 2009)	Fontana, R. M., & Neto, A. I. (2009). ERP systems implementation in complex organizations. *JISTEM - Journal of Information Systems and Technology Management (Online), 6*(1), 61-92. doi: 10.4301/10.4301%2FS1807-17752009000100004

Paper ID	Search	Paper	Full reference
32	4	(Françoise, Bourgault, & Pellerin, 2009)	Françoise, O., Bourgault, M., & Pellerin, R. (2009). ERP implementation through critical success factors' management. *Business Process Management Journal, 15*(3), 371-394. doi: 10.1108/14637150910960620
33	3	(Frimpon, 2012)	Frimpon, M. F. (2012). A Project Approach to Enterprise Resource Planning Implementation. *International Journal of Business and Management, 7*(10), 116. doi: 10.5539/ijbm.v7n10p116
34	3	(Gadakh & Mawale, 2011)	Gadakh, V. S., & Mawale, M. B. (2011). Enterprise Resource Planning (ERP) of Canteen using USA Principle: A case study. *International Journal of Applied Engineering Research, Dindigul, 2*(1), 183-189.
35	4	(Garg & Agarwal, 2014)	Garg, P., & Agarwal, D. (2014). Critical success factors for ERP implementation in a Fortis hospital: an empirical investigation. *Journal of Enterprise Information Management, 27*(4), 402-423. doi: 10.1108/JEIM-06-2012-0027
36	3	(Ghosh & Skibniewski, 2010)	Ghosh, S., & Skibniewski, M. J. (2010). Enterprise Resource Planning Systems Implementation as a Complex Project: A Conceptual Framework. *Journal of Business Economics and Management*, 533-549.
37	3	(Gilfillan & Fowler, 2003)	Gilfillan, M., & Fowler, A. (2003). A Framework for Stakeholder Integration in Higher Education Information Systems Projects. *Technology Analysis & Strategic Management, 15*(4), 468-489. doi: 10.1080/095373203000136051
38	3	(Goossenaerts, Zegers, & Smits, 2009)	Goossenaerts, J. B. M., Zegers, A. T. M., & Smits, J. M. (2009). A multi-level model-driven regime for value-added tax compliance in ERP systems. *Computers in Industry, 60*(9), 709-727. doi: 10.1016/j.compind.2009.05.013
39	4	(Grabski & Leech, 2007)	Grabski, S. V., & Leech, S. A. (2007). Complementary controls and ERP implementation success. *International journal of accounting information systems, 8*(1), 17-39. doi: 10.1016/j.accinf.2006.12.002
40	3	(Griend & Kusters, 2012)	Griend, v. d. P., & Kusters, R. J. (2012). Integration test effort in SAP R/3 systems. *Journal of*

Paper ID	Search	Paper	Full reference
			Software : Evolution and Process, 24(4), 421-435. doi: 10.1002/smr.546
41	6	(Haynes, 2006)	Haynes, R. (2006). *Design Knowledge as a Learning Resource.* Paper presented at the First International Conference on Design Science Research in Information Systems and Technology, February.
42	3	(Helo, Anussornnitisarn, & Phusavat, 2008)	Helo, P., Anussornnitisarn, P., & Phusavat, K. (2008). Expectation and reality in ERP implementation: consultant and solution provider perspective. *Industrial Management & Data Systems, 108*(8), 1045-1059. doi: 10.1108/02635570810904604
43	3	(Holsapple, Wang, & Wu, 2005)	Holsapple, C. W., Wang, Y.-M., & Wu, J.-H. (2005). Empirically Testing User Characteristics and Fitness Factors in Enterprise Resource Planning Success. *International Journal of Human-Computer Interaction, 19*(3), 325-342. doi: 10.1207/s15327590ijhc1903_3
44	4	(Huin, 2004)	Huin, S. F. (2004). Managing deployment of ERP systems in SMEs using multi-agents. *International Journal of Project Management, 22*(6), 511-517. doi: 10.1016/j.ijproman.2003.12.005
45	5	(D. W. Hwang, Min, & Chong, 2015)	Hwang, D. W., Min, H., & Chong, A. (2015). Identifying the drivers of enterprise resource planning and assessing its impacts on supply chain performances. *Industrial Management & Data Systems, 115*(3).
46	3	(Kanchymalay et al., 2013)	Kanchymalay, K., Krishnan, R., Arif, F., Amiruddin, S., Salam, S., & Hashim, U. R. (2013). The Extent of ERP Customization towards User Satisfaction in Daily Operation for Manufacturing Companies. *8*(7), 1788-1792. www.summon.com doi:10.4304/jcp.8.7.1788-1792
47	3	(Kanungo & Bagchi, 2000)	Kanungo, S., & Bagchi, S. (2000). Understanding User Participation and Involvement in ERP Use. *Journal of Management Research, 1*(1), 47.
48	3	(Karimi, Somers, & Bhattacherjee, 2007)	Karimi, J., Somers, T. M., & Bhattacherjee, A. (2007). The Impact of ERP Implementation on Business Process Outcomes: A Factor-Based

Paper ID	Search	Paper	Full reference
			Study. *Journal of Management Information Systems, 24*(1), 101-134. doi: 10.2753/MIS0742-1222240103
49	5	(Khalil, Dominic, & Fadzil bin Hassan, 2010)	Khalil, T., Dominic, P., & Fadzil bin Hassan, M. (2010). *Decision support system framework for implementation of Enterprise Resource Planning (ERP) system.* Paper presented at the Information Technology (ITSim), 2010 International Symposium in.
50	5	(Kien & Lian, 2009)	Kien, S. S., & Lian, Y. P. (2009). Building Enterprise Integration Through Enterprise Resource Planning Systems. *ICIS 2009 Proceedings*, 18.
51	3	(Kovács & Paganelli, 2003)	Kovács, G. L., & Paganelli, P. (2003). A planning and management infrastructure for large, complex, distributed projects—beyond ERP and SCM. *Computers in Industry, 51*(2), 165-183. doi: 10.1016/S0166-3615(03)00034-4
52	3	(Kræmmergaard & Rose, 2002)	Kræmmergaard, P., & Rose, J. (2002). Managerial Competences for ERP Journeys. *Information Systems Frontiers, 4*(2), 199-211. doi: 10.1023/A:1016054904008
53	3	(Kumar, Kumar, & Maheshwari, 2002)	Kumar, V., Kumar, U., & Maheshwari, B. (2002). ERP systems implementation: best practices in Canadian government organizations. *Government Information Quarterly, 19*(2), 147-172. doi: 10.1016/S0740-624X(02)00092-8
54	3	(O. B. Kwon & Lee, 2001)	Kwon, O. B., & Lee, J. J. (2001). A multi-agent intelligent system for efficient ERP maintenance. *Expert Systems with Applications, 21*(4), 191-202. doi: 10.1016/S0957-4174(01)00039-2
55	4	(Larsen, 2009)	Larsen, T. J. (2009). A multilevel explanation of end-user computing satisfaction with an enterprise resource planning system within an international manufacturing organization. *Computers in Industry, 60*(9), 657-668. doi: 10.1016/j.compind.2009.05.004
56	4	(J. C. Lee & Myers, 2004)	Lee, J. C., & Myers, M. D. (2004). Dominant actors, political agendas, and strategic shifts over time: a critical ethnography of an enterprise systems implementation. *Journal of Strategic Information Systems, 13*(4), 355-374. doi:

Paper ID	Search	Paper	Full reference
			10.1016/j.jsis.2004.11.005
57	3	(Lengnick-Hall, Lengnick-Hall, & Abdinnour-Helm, 2004)	Lengnick-Hall, C. A., Lengnick-Hall, M. L., & Abdinnour-Helm, S. (2004). The role of social and intellectual capital in achieving competitive advantage through enterprise resource planning (ERP) systems. *Journal of Engineering and Technology Management, 21*(4), 307-330. doi: 10.1016/j.jengtecman.2004.09.005
58	3	(Li, Liao, & Lei, 2006)	Li, Y., Liao, X. W., & Lei, H. Z. (2006). A knowledge management system for ERP implementation. *Systems Research and Behavioral Science, 23*(2), 157-168. doi: 10.1002/sres.751
59	3	(Light et al., 2001)	Light, B., Holland, C. P., & Wills, K. (2001). ERP and best of breed: a comparative analysis. *Business Process Management Journal, 7*(3), 216-224. doi: 10.1108/14637150110392683
60	4	(Little & Best, 2003)	Little, A., & Best, P. J. (2003). A framework for separation of duties in an SAP R/3 environment. *Managerial Auditing Journal, 18*(5), 419-430. doi: 10.1108/02686900310476882
61	3	(Lungu & Vãtuiu, 2007)	Lungu, I. O. N., & VĂTuiu, T. (2007). COMPUTER ASSISTED AUDIT TECHNIQUES. *Annals of the University of Petrosani : Economics*, 217-224.
62	5	(Lyytinen, Newman, & Al-Muharfi, 2009)	Lyytinen, K., Newman, M., & Al-Muharfi, A.-R. A. (2009). Institutionalizing enterprise resource planning in the Saudi steel industry: a punctuated socio-technical analysis. *Journal of Information Technology, 24*(4), 286-304.
63	4	(M. Beheshti, K. Blaylock, A. Henderson, & G. Lollar, 2014)	M. Beheshti, H., K. Blaylock, B., A. Henderson, D., & G. Lollar, J. (2014). Selection and critical success factors in successful ERP implementation. *Competitiveness review, 24*(4), 357-375. doi: 10.1108/CR-10-2013-0082
64	4	(Maguire, Ojiako, & Said, 2010)	Maguire, S., Ojiako, U., & Said, A. (2010). ERP implementation in Omantel: a case study. *Industrial Management & Data Systems, 110*(1), 78-92. doi: 10.1108/02635571011008416
65	3	(Maheshwari, Kumar, & Kumar, 2010)	Maheshwari, B., Kumar, V., & Kumar, U. (2010). Delineating the ERP institutionalization process: go-live to effectiveness. *Business Process Man-*

Paper ID	Search	Paper	Full reference
			agement Journal, 16(4), 744-771. doi: 10.1108/14637151011065982
66	3	(M. Lynne Markus, Tanis, & van Fenema, 2000)	Markus, M. L., Tanis, C., & van Fenema, P. C. (2000). Multisite ERP implementations (Vol. 43, pp. 42): H.W. Wilson - Applied Science & Technology Abstracts.
67	3	(Marler & Liang, 2012)	Marler, J. H., & Liang, X. (2012). Information technology change, work complexity and service jobs: a contingent perspective. *New Technology, Work and Employment, 27*(2), 133-146. doi: 10.1111/j.1468-005X.2012.00280.x
68	3	(Marnewick & Labuschagne, 2005)	Marnewick, C., & Labuschagne, L. (2005). A conceptual model for enterprise resource planning (ERP). *Information Management & Computer Security, 13*(2), 144-155. doi: 10.1108/09685220510589325
69	3	(Martin, Mauterer, & Gemunden, 2002)	Martin, R., Mauterer, H., & Gemunden, H. G. (2002). Classifying the benefits of ERP systems in the manufacturing industry. *WIRTSCHAFTSINFORMATIK, 44*(2), 109-116.
70	4	(Masbahi, Bahsani, & Semma, 2012)	Masbahi, M. E., Bahsani, S., & Semma, A. (2012). Enterprise Resource Planning (ERP) and Integration of Purchase, Inventory and Sale processes: Proposition of a Common Minimal Model. *International Journal of Computer Science Issues (IJCSI), 9*(5), 181-190.
71	3	(Masini & Wassenhove, 2009)	Masini, A., & Wassenhove, L. N. (2009). ERP competence-building mechanisms: an exploratory investigation of configurations of ERP adopters in the European and US manufacturing sectors. *Manufacturing & service operations management, 11*(2), 274-298. doi: 10.1287/msom.1080.0215
72	3	(McAdam & Galloway, 2005)	McAdam, R., & Galloway, A. (2005). Enterprise resource planning and organisational innovation: a management perspective. *Industrial Management & Data Systems, 105*(3), 280-290. doi: 10.1108/02635570510590110
73	2	(Miltenburg, 2001)	Miltenburg, J. (2001). Computational complexity of algorithms for MRP and JIT production planning problems in enterprise resource planning

Paper ID	Search	Paper	Full reference
			systems. *Production Planning & Control, 12*(2), 198-209. doi: 10.1080/09537280150501293
74	3	(Ming-Chang Lee, 2011)	Ming-Chang Lee, J.-F. C. J.-F. C. (2011). An Entropy Decision Model for Selection of Enterprise Resource Planning System. *International Journal of Computer Trends and Technology, 1*, 162-169.
75	3	(Charles Møller, 2005)	Møller, C. (2005). ERP II: a conceptual framework for next-generation enterprise systems? *Journal of Enterprise Information Management, 18*(4), 483-497. doi: 10.1108/17410390510609626
76	4	(Nandhakumar, Rossi, & Talvinen, 2005)	Nandhakumar, J., Rossi, M., & Talvinen, J. (2005). The dynamics of contextual forces of ERP implementation. *Journal of Strategic Information Systems, 14*(2), 221-242. doi: 10.1016/j.jsis.2005.04.002
77	5	(Newell, Huang, Galliers, & Pan, 2003)	Newell, S., Huang, J. C., Galliers, R. D., & Pan, S. L. (2003). Implementing enterprise resource planning and knowledge management systems in tandem: fostering efficiency and innovation complementarity. *Information and Organization, 13*(1), 25-52.
78	4	(Ojiako, Papadopoulos, Thumborisuthi, & Fan Yang, 2012)	Ojiako, U., Papadopoulos, T., Thumborisuthi, C., & Fan Yang, Y. (2012). Perception variability for categorised risk factors. *Industrial Management & Data Systems, 112*(4), 600-618. doi: 10.1108/02635571211225503
79	5	(Özen, 2011)	ÖZEN, M. (2011). Product Structure Modeling in an Enterprise Resource Planning System.
80	3	(Palanisamy, 2007)	Palanisamy, R. (2007). Organizational culture and knowledge management in ERP implementation: An empirical study. *JOURNAL OF COMPUTER INFORMATION SYSTEMS, 48*(2), 100-120.
81	3	(Pan & Jang, 2008)	Pan, M.-J., & Jang, W.-Y. (2008). DETERMINANTS OF THE ADOPTION OF ENTERPRISE RESOURCE PLANNING WITHIN THE TECHNOLOGY-ORGANIZATION-ENVIRONMENT FRAMEWORK: TAIWAN'S COMMUNICATIONS INDUSTRY. *The Journal of Computer Information Systems, 48*(3), 94.

Paper ID	Search	Paper	Full reference
82	6	(Parrish, 2008)	PARRISH JR, J. L. (2008). Sensemaking in information systems: toward a sensemaking inquiring system.
83	4	(Poba-Nzaou, Raymond, & Fabi, 2008)	Poba-Nzaou, P., Raymond, L., & Fabi, B. (2008). Adoption and risk of ERP systems in manufacturing SMEs: a positivist case study. *Business Process Management Journal, 14*(4), 530-550. doi: 10.1108/14637150810888064
84	3	(Ram, Corkindale, & Wu, 2013)	Ram, J., Corkindale, D., & Wu, M. L. (2013). ENTERPRISE RESOURCE PLANNING ADOPTION: STRUCTURAL EQUATION MODELING ANALYSIS OF ANTECDANTS. *JOURNAL OF COMPUTER INFORMATION SYSTEMS, 54*(1), 53-65.
85	4	(Ranganathan & Brown, 2006)	Ranganathan, C., & Brown, C. V. (2006). ERP Investments and the Market Value of Firms: Toward an Understanding of Influential ERP Project Variables. *Information Systems Research, 17*(2), 145-161. doi: 10.1287/isre.1060.0084
86	3	(Rose & Schlichter, 2013)	Rose, J., & Schlichter, B. R. (2013). Decoupling, re-engaging: managing trust relationships in implementation projects. *Information Systems Journal, 23*(1), 5-33. doi: 10.1111/j.1365-2575.2011.00392.x
87	3	(Ruivo et al., 2014)	Ruivo, P., Oliveira, T., & Neto, M. (2014). Examine ERP post-implementation stages of use and value: empirical evidence from Portuguese SMEs. *International journal of accounting information systems, 15*(2), 166-184. doi: 10.1016/j.accinf.2014.01.002
88	3	(Jose L. Salmeron & Lopez, 2010)	Salmeron, J. L., & Lopez, C. (2010). A multicriteria approach for risks assessment in ERP maintenance. *The Journal of Systems & Software, 83*(10), 1941-1953. doi: 10.1016/j.jss.2010.05.073
89	4	(J. L. Salmeron & Lopez, 2012)	Salmeron, J. L., & Lopez, C. (2012). Forecasting Risk Impact on ERP Maintenance with Augmented Fuzzy Cognitive Maps. *IEEE Transactions on Software Engineering, 38*(2), 439-452. doi: 10.1109/TSE.2011.8

Paper ID	Search	Paper	Full reference
90	3	(Sammon & Adam, 2005)	Sammon, D., & Adam, F. (2005). Towards a model of organisational prerequisites for enterprise-wide systems integration: Examining ERP and data warehousing. *Journal of Enterprise Information Management, 18*(4), 458-470. doi: 10.1108/17410390510609608
91	6	(Shafiei-Monfared & Jenab, 2012)	Shafiei-Monfared, S., & Jenab, K. (2012). Fuzzy Complexity Model for Enterprise Maintenance Projects. *Engineering Management, IEEE Transactions on, 59*(2), 293-298.
92	4	(Soliman & Youssef, 2001)	Soliman, F., & Youssef, M. (2001). The impact of some recent developments in e-business on the management of next generation manufacturing. *International journal of operations & production management, 21*(5/6), 538-564. doi: 10.1108/01443570110390327
93	3	(Sommer, 2003)	Sommer, R. A. (2003). Business process flexibility: a driver for outsourcing. *Industrial Management & Data Systems, 103*(3), 177-183. doi: 10.1108/02635570310465652
94	9	(Spiteri et al., 2012)	Spiteri, K. J., Luca, C. L., Reynolds, T., & Wilson, G. (2012). Defining a baseline complexity model for ERP systems over SaaS. *Journal of Internet Technology and Secured Transactions*. Spiteri, K. J., Luca, C. L., Reynolds, T., & Wilson, G. Defining a baseline complexity model for ERP systems over SaaS. *Journal of Internet Technology and Secured Transactions.*
95	3	(Štemberger, Vukšić, & Kovačič, 2009)	Štemberger, M. I., Vukšić, V. B., & Kovačič, A. (2009). Business Process Modelling as a Critical Success Factor in Implementing an ERP System. *South East European Journal of Economics and Business, 4*(2), 89-96. doi: 10.2478/v10033-009-0014-3
96	3	(Strong & Volkoff, 2004)	Strong, D. M., & Volkoff, O. (2004). A roadmap for enterprise system implementation. *Computer, 37*(6), 22-29. doi: 10.1109/MC.2004.3
97	3	(Sullivan & Porter, 2006)	Sullivan, L., & Porter, R. (2006). Implementing student information systems. *New Directions for Higher Education, 2006*(136), 35-51. doi:

Paper ID	Search	Paper	Full reference
			10.1002/he.238
98	5	(Taskin, 2011)	Taskin, N. (2011). *Flexibility and strategic alignment of enterprise resource planning systems with business strategies: an empirical study.*
99	3	(Tejeida-Padilla, Badillo-Pina, & Morales-Matamoros, 2010)	Tejeida-Padilla, R., Badillo-Pina, I., & Morales-Matamoros, O. (2010). A Systems Science Approach to Enterprise Resources Planning Systems. *Systems Research and Behavioral Science, 27*(1), 87-95. doi: 10.1002/sres.957
100	3	(Teltumbde, 2000)	Teltumbde, A. (2000). A framework for evaluating ERP projects. *International Journal of Production Research, 38*(17), 4507-4520. doi: 10.1080/00207540050205262
101	6	(Tomkos, 1999)	Tomkos, T. (1999). *Formal specification and modeling of complex systems.*
102	3	(Umble et al., 2003)	Umble, M. M., Umble, E. J., & Haft, R. R. (2003). Enterprise resource planning: Implementation procedures and critical success factors. *European Journal of Operational Research, 146*(2), 241-257. doi: 10.1016/S0377-2217(02)00547-7
103	3	(Verville & Halingten, 2002)	Verville, J., & Halingten, A. (2002). An investigation of the decision process for selecting an ERP software: the case of ESC. *Management Decision, 40*(3), 206-216. doi: 10.1108/00251740210420156
104	6	(Viskoviü & Varga, 2008)	Viskoviü, D., & Varga, M. (2008). Bad practices in complex IT projects.
105	3	(Vosburg & Kumar, 2001)	Vosburg, J., & Kumar, A. (2001). Managing dirty data in organizations using ERP: lessons from a case study. *Industrial Management & Data Systems, 101*(1), 21-31. doi: 10.1108/02635570110365970
106	6	(Wang, Hsieh, & Butler, 2006)	Wang, W., Hsieh, J., & Butler, J. E. (2006). System emergent use: a theoretical model and empirical exploration.
107	4	(Ward, Hemingway, & Daniel, 2005)	Ward, J., Hemingway, C., & Daniel, E. (2005). A framework for addressing the organisational issues of enterprise systems implementation. *Journal of Strategic Information Systems, 14*(2), 97-119. doi: 10.1016/j.jsis.2005.04.005

Paper ID	Search	Paper	Full reference
108	4	(J. Williams, Williams, & Morgan, 2013)	Williams, J., Williams, M. D., & Morgan, A. (2013). A teleological process theory for managing ERP implementations. *Journal of Enterprise Information Management, 26*(3), 235-249. doi: 10.1108/17410391311325216
109	3	(Wood & Caldas, 2001)	Wood, T., & Caldas, M. P. (2001). Reductionism and complex thinking during ERP implementations. *Business Process Management Journal, 7*(5), 387-393. doi: 10.1108/14637150110406777
110	3	(Worley, Chatha, Weston, Aguirre, & Grabot, 2005)	Worley, J. H., Chatha, K. A., Weston, R. H., Aguirre, O., & Grabot, B. (2005). Implementation and optimisation of ERP systems: A better integration of processes, roles, knowledge and user competencies. *Computers in Industry, 56*(6), 620-638. doi: 10.1016/j.compind.2005.03.006
111	3	(F. Wu, Li, Chu, Sculli, & Gao, 2009)	Wu, F., Li, H. Z., Chu, L. K., Sculli, D., & Gao, K. (2009). An approach to the valuation and decision of ERP investment projects based on real options. *Annals of operations research, 168*(1), 181-203. doi: 10.1007/s10479-008-0365-7
112	3	(J.-H. Wu & Wang, 2007)	Wu, J.-H., & Wang, Y.-M. (2007). Measuring ERP success: The key-users' viewpoint of the ERP to produce a viable IS in the organization. *Computers in Human Behavior, 23*(3), 1582-1596. doi: 10.1016/j.chb.2005.07.005
113	3	(W.-W. Wu, Lee, & Lan, 2013)	Wu, W.-W., Lee, Y.-T., & Lan, L. W. (2013). On the Decision Structures and Knowledge Discovery for ANP Modeling. *International Journal of Intelligence Science, 3*, 15-23. doi: 10.4236/ijis.2013.31A003
114	3	(Xu, Wang, & Luo, 2006)	Xu, L., Wang, C., & Luo, X. (2006). Integrating Knowledge Management and ERP in Enterprise Information Systems. *Systems Research and Behavioral Science [H.W.Wilson - SSA], 23*(2), 147.
115	3	(Xue, Liang, Boulton, & Snyder, 2005)	Xue, Y., Liang, H., Boulton, W. R., & Snyder, C. A. (2005). ERP implementation failures in China: Case studies with implications for ERP vendors. *International Journal of Production Economics, 97*(3), 279-295. doi: 10.1016/j.ijpe.2004.07.008

Paper ID	Search	Paper	Full reference
116	3	(Yahia, Aubry, & Panetto, 2012)	Yahia, E., Aubry, A., & Panetto, H. (2012). Formal measures for semantic interoperability assessment in cooperative enterprise information systems. *Computers in Industry, 63*(5), 443-457. doi: 10.1016/j.compind.2012.01.010
117	3	(Yeh & OuYang, 2010)	Yeh, J. Y., & OuYang, Y.-c. (2010). How an organization changes in ERP implementation: a Taiwan semiconductor case study. *Business Process Management Journal, 16*(2), 209-225. doi: 10.1108/14637151011035561

Appendix 3.4: Rating papers about complexity and ERP implementations

		Statements				Characteristics				
		ERP systems are complex	ERP projects are complex	ERP implementations are complex	ERP projects have complex environments	Occurring of the term "complexity" related to ERP implementations	Occurring of the term "complexity" unrelated to ERP implementations	Term complexity not found in main paper text	Complexity theory is mentioned	The term complexity is discussed
Paper ID	Search	**50**	**29**	**35**	**15**	**66**	**89**	**7**	**7**	**5**
1	3	X					X			
2	3							X		
3	4		X	X		X				
4	3	X	X				X			
5	3	X	X		X		X			
6	3	X								
7	3		X	X			X			
8	3	X					X			
9	3						X			
10	4	X	X	X		X				
11	3			X		X	X			
12	3			X		X				
13	5	X		X			X			
14	5	X					X			X
15	4						X			
16	3	X				X	X			
17	3	X		X		X	X			X
18	4						X			
19	3	X				X				
20	3	X		X	X	X	X			
21	4		X			X				
22	6						X			
23	3						X			
24	4		X			X	X			

		Statements				Characteristics				
		ERP systems are complex	ERP projects are complex	ERP implementations are complex	ERP projects have complex environments	Occurring of the term "complexity" related to ERP implementations	Occurring of the term "complexity" unrelated to ERP implementations	Term complexity not found in main paper text	Complexity theory is mentioned	The term complexity is discussed
Paper ID	Search	**50**	**29**	**35**	**15**	**66**	**89**	**7**	**7**	**5**
25	4							X		
26	3			X		X				
27	3			X	X	X	X			
28	3	X		X		X	X			
29	4	X	X		X	X				
30	3	X				X	X			
31	3						X		X	X
32	4		X			X	X			
33	3			X		X	X			
34	3						X			
35	4	X	X	X	X	X				
36	3		X	X	X	X				X
37	3	X				X	X			
38	3						X			
39	4	X	X	X	X	X	X			
40	3	X				X	X			
41	6						X			
42	3	X		X		X	X			
43	3	X		X		X				
44	4		X			X	X			
45	5			X			X			
46	3						X			
47	3	X				X	X			
48	3			X		X	X			
49	5									
50	5			X			X			
51	3						X			
52	3			X		X	X			

		Statements				Characteristics				
		ERP systems are complex	ERP projects are complex	ERP implementations are complex	ERP projects have complex environments	Occurring of the term "complexity" related to ERP implementations	Occurring of the term "complexity" unrelated to ERP implementations	Term complexity not found in main paper text	Complexity theory is mentioned	The term complexity is discussed
Paper ID	Search	**50**	**29**	**35**	**15**	**66**	**89**	**7**	**7**	**5**
53	3	X	X	X		X				
54	3	X				X				
55	4	X	X			X	X			
56	4	X	X			X				
57	3	X				X			X	
58	3			X		X				
59	3			X		X	X			
60	4							X		
61	3	X				X	X			
62	5	X		X	X		X			
63	4	X	X			X				
64	4		X	X		X				
65	3	X		X		X	X			
66	3						X			
67	3						X			
68	3	X				X				
69	3	X				X	X			
70	4	X	X			X	X			
71	3	X			X	X	X			
72	3						X			
73	2						X			
74	3				X	X	X			
75	3		X			X	X			
76	4				X	X	X			
77	5						X			
78	4							X		
79	5	X					X			
80	3	X		X		X	X			

		Statements				Characteristics				
		ERP systems are complex	ERP projects are complex	ERP implementations are complex	ERP projects have complex environments	Occurring of the term "complexity" related to ERP implementations	Occurring of the term "complexity" unrelated to ERP implementations	Term complexity not found in main paper text	Complexity theory is mentioned	The term complexity is discussed
Paper ID	Search	**50**	**29**	**35**	**15**	**66**	**89**	**7**	**7**	**5**
81	3	X				X	X			
82	6						X			
83	4						X		X	
84	3	X	X			X	X			
85	4		X			X	X			
86	3						X			
87	3	X					X			
88	3	X	X			X	X			
89	4		X			X	X			
90	3	X		X		X	X			
91	6						X			
92	4						X		X	
93	3							X		
94	9	X				X	X		X	X
95	3		X		X	X	X			
96	3						X			
97	3	X		X		X	X			
98	5	X					X		X	
99	3						X			
100	3		X		X	X	X			
101	6						X			
102	3	X				X	X			
103	3	X	X			X	X			
104	6						X			
105	3				X	X	X			
106	6						X			
107	4			X		X	X			
108	4	X	X	X	X		X			

		Statements				Characteristics				
		ERP systems are complex	ERP projects are complex	ERP implementations are complex	ERP projects have complex environments	Occurring of the term "complexity" related to ERP implementations	Occurring of the term "complexity" unrelated to ERP implementations	Term complexity not found in main paper text	Complexity theory is mentioned	The term complexity is discussed
Paper ID	Search	**50**	**29**	**35**	**15**	**66**	**89**	**7**	**7**	**5**
109	3		X	X		X		X	X	
110	3			X		X	X			
111	3							X		
112	3	X				X				
113	3						X			
114	3						X			
115	3	X				X	X			
116	3						X			
117	3			X		X	X			

Appendices Chapter 4

Appendix 4.1: used papers with lists of activities within ERP implementation projects

Number	Paper	Category
1.	Al-Mashari, M., Al-Mudimigh, A., & Zairi, M. (2003). Enterprise resource planning: A taxonomy of critical factors. European Journal of Operational Research, 146(2), 352-364.	A
2.	Berchet, C., & Habchi, G. (2005). The implementation and deployment of an ERP system: An industrial case study. 56(6), 588-605.	B
3.	Bruges, P. (2002). ERP Implementation Methodologies. MSIS, 488.	C
4.	Ehie, I. C., & Madsen, M. (2005). Identifying critical issues in enterprise resource planning (ERP) implementation. 56(6), 545-557.	A
5.	Esteves, J., & Pastor, J. A. (2001). Analysis of critical success factors relevance along SAP implementation phases. Seventh Americas Conference on Information Systems.	A
6.	Francalanci, C. (2001). Predicting the implementation effort of ERP projects: empirical evidence on SAP/R3. Journal of Information Technology, Volume 16(1), 33 - 48.	A
7.	Hallikainen, P., Kimpimäki, H., & Kivijärvi, H. (2006). Supporting the Module Sequencing Decision in the ERP Implementation Process. Proceedings of the 39th Hawaii International Conference on System Sciences - 2006, 1-10.	A
8.	Kumar, V., Maheshwari, B., & Kumar, U. (2003). An investigation of critical management issues in ERP implementation: empirical evidence from Canadian organizations. Technovation, 23(10), 793-807.	A
9.	Latvanen, H., & Ruusunen, R. (2001). Management of Risks in an ERP Implementation Project. In T. S. o. Economics (Ed.), (pp. 20).	A
10.	Mabert, V. A., Soni, A., & Venkataramanan, M. A. (2005). Model based interpretation of survey data: A case study of enterprise resource planning implementations. In Press, Corrected Proof.	A
11.	Markus, M. L., & Tanis, C. (2003). The Enterprise System Experience - From Adoption to Success. Pinnaflex Educational Resources 173-207.	A
12.	Marnewick, C., & Labuschagne, L. (2005). A conceptual model for enterprise resource planning (ERP). Information Management & Computer Security, 13(2), 144-155.	A
13.	Parr, A., & Shanks, G. (2000). A model of ERP project implementation. Journal of Information Technology, 15(4), 289-303.	A
14.	Rajagopal, P. (2002). An innovation--diffusion view of implementation of enterprise resource planning (ERP) systems and development of a research model. Information & Management, 40(2), 87-114.	A
15.	Sarker, S., & Lee, A. S. (2003). Using a case study to test the role of three key social enablers in ERP implementation. Information & Management, 40(8), 813-829.	B
16.	Somers, T. M., & Nelson, K. G. (2004). A taxonomy of players and activities across the ERP project life cycle. Information & Management, 41(3), 257-278.	A
17.	Sumner, M. (2000). Risk factors in enterprise-wide/ERP projects. Journal of Information Technology, 15(4).	A
18.	Tchokogué, A., Bareil, C., & Duguay, C. R. (2005). Key lessons from the implementation of an ERP at Pratt & Whitney Canada. International Journal of Production Economics, In Press, Corrected Proof.	B
19.	Umble, E. J., Haft, R. R., & Umble, M. M. (2003). Enterprise resource planning: Implementation procedures and critical success factors. European Journal of Operational Research, 146(2), 241-257.	A
20.	Wagner, W., & Antonucci, Y. L. (2004). An analysis of the imagine PA public sector ERP project. System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on, 8.	A
21.	Wei, C.-C., Chien, C.-F., & Wang, M.-J. J. (2005). An AHP-based approach to ERP system selection. International Journal of Production Economics, In Press, Corrected Proof.	A
22.	Wei, C.-C., & Wang, M.-J. J. (2004). A comprehensive framework for selecting an ERP system. International Journal of Project Management, 22(2), 161-169.	A
23.	Weston, F. C. W. J. (2001). ERP implementation and project management. Production and Inventory Management Journal, 42(3/4), 75.	A
24.	Yusuf, Y., Gunasekaran, A., & Abthorpe, M. S. (2004). Enterprise information systems project implementation: A case study of ERP in Rolls-Royce. International Journal of Production Economics, 87(3), 251-266.	B

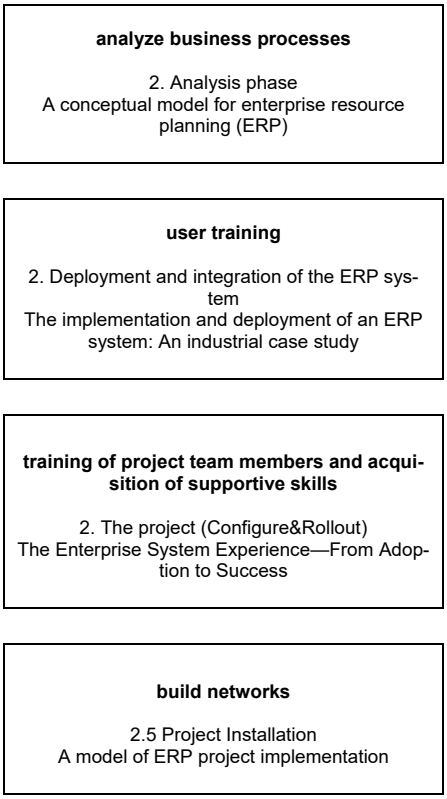
Legend:

A = Papers which relate risk factors and Critical Success Factors (CSF's) or other influencing factors to activities and/or project phases.

B = Papers about cases which describe the phases and activities of the actual projects.

C = Papers which describe standard project phases and activities from consultancy firms or ERP software suppliers.

Appendix 4.2: Examples of stickers



Appendix 4.3: Clusters And Subclusters With Activities And references

Cluster	Subcluster	Activity Number	Activity	Reference number from appendix A
Selection				
	Vendor selection			
		1	Select consulting company	3
		2	Selecting implementation partner	9
		3	Establish contracts	2, 3, 19
		4	Interview vendors and collect detailed information	22
	Product selection			
		5	Select ERP vendor	14, 23
		6	Identify the ERP system characteristics	21
		7	Choose appropriate technology	14
		8	Define expression of requirements and specifications	2
		9	Create the request for proposal (RFP)	19
		10	Construct the structure of objectives	21
		11	Create a software candidate list	19
		12	Analyse functionality, price, training and maintenance services	3, 19, 21, 22, 23
		13	Analyze current Business Processes and selecting ERP-system	4
		14	Select ERP package	1, 3, 14, 16, 19, 22
		15	Define contractual agreement	3
		16	Produce request for proposal (to vendors software)	23
		17	Make evaluation scheme for comparing and ranking vendor responses	23
		18	Check references ERP vendors	23
		19	Selection of ERP product, project manager and implementation partners	8
		20	Collect all possible information about ERP vendors and systems. Filter out unqualified vendors	22
Project configuration				
		21	Develop project plan	1, 3, 4, 5, 7, 8, 9, 11, 13, 17, 18, 23, 24
		22	Compose team	3, 4, 5, 7, 8, 9, 11, 12, 13, 16, 21, 24
		23	Select the project leader	9
		24	Form steering committee	7, 9, 13, 24
		25	Format budget	2, 9, 24
		26	Develop project deliverables	23
		27	Define project objectives	3
		28	Define areas of responsibility	23
		29	Develop project charter	23

		30	Plan for project reviews	23
		31	Planning Variables	10
		32	Scoping & Planning	20
		33	Address change control procedures	23
		34	Address planning and implementation tools	23
		35	Development of the project's guiding principles	7
		36	Decision to proceed, approval of project plan	11
		37	Reporting mechanisms	13
		38	Develop metrics (for revenues implementation ERP system)	23
		39	Address tools to measure performance results	21, 23
Project manage age-ment				
	Management			
		40	Ongoing project management	6, 11, 16
		41	Interdepartmental cooperation	16
		42	Proceed planning	23
		43	Change management	16
	Communication to organization			
		44	Interdepartmental communication	16
		45	Communication to organization	11
		46	Constant communication with users	13
		47	Create communication plan	1
Organi-zational and system design				
	Current state analy-sis			
		48	Current state analysis (may be deferred or not done)	11
		49	Analyse current business processes	7, 13
		50	Map business processes on to ERP functions	7
		51	Evaluate processes in place	18
		52	Analyse organizational processes and compare them with the procedures embedded in the ERP package	6
	Organizational requirements			
		53	Business process reengineering	1, 4, 5, 16, 18, 22
		54	Identify process redesign	12
		55	Current and/or future business process modelling and reengi-neering, if any	11
		56	High level design review (analyse the enterprise model, and develop ' Vanilla' prototype)	24
		57	Develop metrics (for revenues implementation ERP system)	23
		58	Address tools to measure performance results	21, 23
		59	Develop initial audit procedures	3
	Requirements ERP system			
		60	Definition of system requirements	3, 9, 12, 13

		61	Identify operational needs	12
		62	Review functional and technical requirements to determine the system build needs	12
		63	Requirements analysis	17
		64	Create a feature/function list	19
		65	Identifying modules needed	9
		66	Determine the software components of the ERP system	12
		67	Finalize requirement definition stage (scope, schedule, resource requirements quality concerns, risk concerns, organizational issues	23
	High level design			
		68	Define business processes (blueprint)	3, 5
		69	High-level design	2, 7, 9, 12, 13, 20
		70	Evaluate alternatives to comprehensive engineering project	3
		71	Craf "best-fit" approach	3
		72	Business processes into ERP system	9
		73	Preliminary design review (developing a design and implementation strategy, defining the scope of the project, and developing the business process model)	24
Configuration and installation				
	System configuration			
		74	Systems design	7, 13, 17
		75	Customize and parameterization of ERP software	3, 5, 6, 11, 18
		76	Install ERP	7, 13, 14
		77	High-level design	13
		78	Configure baseline system	3
		79	Mastering ERP system (functionality, configuration)	4
		80	Development of a comprehensive configuration	7
		81	Identify functionality delivery options	12
		82	Configurator implementation	15
		83	Starting preparation	2
		84	Reduce the number of specific programs	2
		85	Develop comprehensive configuration	13
		86	Write and test reports	13
		87	Install the software and perform the computer room pilot	19
		88	Establish security and necessary permissions	19
		89	Define the system hierarchy	9
		90	Install prototype system	24
	Data conversion			
		91	Convert data	5
		92	Data analysis and conversion	16
		93	Data cleanup and conversion	11
		94	Transfer data from legacy systems	24
	System integration			

		95	Identify data and system interfaces	9, 13
		96	Build and test interfaces	7, 9, 13
		97	Integrate with other systems	1, 3, 11
		98	Analyze legacy systems	1
		99	Determine the software components of the ERP system interact with each other	12
		100	Integrate functional units	14
		101	Technology integration and implementation	17
		102	Ensure that all data bridges are sufficiently robust and the data are sufficiently accurate	19
		103	Replace legacy systems	24
	ERP system Testing			
		104	Test ERP system	1, 3, 5, 7, 9, 13, 18
		105	Test reports	7, 9
		106	Population of the test instance with real data	7, 13
		107	Test with real data	9
		108	Acceptance test	23
		109	Create test scenario	1
		110	Build & Test	20
		111	Evaluate the process model and the information system build against each other	12
		112	Testing, bug fixing, and rework	11
		113	Critical design review (integration testing)	24
		114	Implementation realisation (user acceptance testing)	24
		115	Technical/operation review (user acceptance testing)	24
		116	Post implementation review (system deployment, systems conversion, user training before the ' Go Live')	24
		117	Test all modules against requirements as well as quality parameters	6
Customizing				
		118	Interactive prototyping	7, 9, 13
		119	Specify functional and data requirements	6
		120	Customization	11, 16
		121	Custom programming and documentation	23
		122	Technical development (modifications, interfaces, data conversion)	4
		123	Detailed design, realisation, and prototype validation	2
		124	Develop and verify software code for modules that need reprogramming	6
Infrastructure				
		125	Establish contracts	2
		126	Hardware acquisition	23
		127	Determine operating system	23
		128	Decide on wireless requirements	23
		129	Plan infrastructure	1
		130	Infrastructure up gradation	8
		131	IT integration at global levels realized	14
		132	Architecture choices	16

		133	Selection of software, hardware platform, networking, database, implementation partner, project manager (may be partially or totally deferred to project phase)	11
		134	Build networks	13
		135	Install desktops	13
		136	Install and test any new hardware	19
		137	Inventory existing hardware and software	9
		138	Investigate incompatibility	14
Reorganization				
		139	Change culture+structure organization	1
		140	Software configuration and "fit with the organization" (Current and/or future business process modelling and reengineering, if any, Execution of change management plan, if any, Software configuration, Software customization if any, System integration, Integration of software bolt-ons and/or legacy systems, if any, Data cleanup and conversion, Documentation	8
		141	Identify change ownership	12
		142	Make final changes to business processes, policies and procedures and system builds tot prepare for a go-live	12
		143	Observe user resistance	14
		144	Realize organizational integration	14
		145	Organizational structure and culture change	15
		146	Organizational changes and/or incentives related to enterprise system and/or organizational performance improvement, if any (may be deferred)	11
		147	Execution of change management plan, if any	11
		148	Process and procedure changes	11
		149	User communications and gaining acceptance	9
System implementation				
		150	Implement	2, 20, 22, 24
		151	Go live	4, 23
		152	Testing, bug fixing, and rework	11
		153	Conference room pilot (prototyping and adjustment toward final system)	4
		154	Rollout and start-up	8
		155	Make systems available for usage	14
		156	Use systems in individual units	14
		157	Increase use of systems	14
		158	Users accept the systems	14
		159	use of systems become a routine activity	14
		160	Correct flaws	14
		161	Implementation of core modules of the selected ERP package	15
		162	Users understand, assimilate and then appropriate their new tool	2
		163	Rollout and start-up	11
		164	Systems implementation/maintenance	17
		165	Run a pre-implementation pilot	19
		166	Review the pre-implementation process to date	19

		167	Bring the entire organization on-line, either in a total cutover or in a phased approach	19
		168	Celebrate	19
		169	Cutover from the old systems	18
		170	Usage of the system is not an 'out of the ordinary' situation	14
Training				
	Training implementation staff			
		171	Train the project team	3, 7, 9, 13
		172	Training of project team members and acquisition of supportive skills	8, 11
	Training users			
		173	Prepare end-user training	5
		174	Begin training activities	14
		175	Train users	1, 2, 3, 5, 8, 11, 16, 23
		176	Educate and train critical mass (on processes, data discipline and modules)	4
		177	Train users more	14
		178	Education on new business processes	16
		179	Managing user training and support	13
		180	Attend system training	19
		181	Train on the conference room pilot	19
	Training maintenance staff			
		182	Problem resolution (adding hardware capacity, process and procedure changes, user acceptance, retraining, additional training)	8
		183	Create support for tangible operational processes and information system	12
Set up maintenance				
		184	Optimize tool	2, 4, 11, 19
		185	Correct malfunctions	3, 11, 13, 23
		186	Fine tune system	3
		187	Enhance original implementation	23
		188	Create additional metrics	23
		189	Go & Live Support	5
		190	Adjust system	3
		191	Use product	3
		192	Maintain product	3
		193	Meet special optimisation requests	3
		194	Tuning and testing (finalize processing options, profiles, menus, and testing robustness)	4
		195	Testing, bug fixing, and rework	8
		196	Challenges (bug fixing, rework, system performance tuning	8
		197	Problem resolution (adding hardware capacity, process and procedure changes, user acceptance, retraining, additional training)	8
		198	Create support for tangible operational processes and information system	12
		199	Modify systems to fit user needs carried out	14

		200	Enhance compatibility	14
		201	Operational starting with production	2
		202	Detect key processes of improvement	2
		203	Start potential modifications	2
		204	Optimise the ERP deployment process itself	2
		205	Adding hardware capacity	11
		206	Retraining, additional training	11
		207	Adding people to accommodate learning and shakedown needs	11
		208	Extension and transformation	13

Appendix 4.4: List of selected scientific papers for ERP activity retrieval

1. Al-Mashari, M., & Al-Mudimigh, A. (2003). ERP implementation: lessons from a case study. *Information Technology, 16*, 21-33.
2. Al-Mashari, M., Al-Mudimigh, A., & Zairi, M. (2003). Enterprise resource planning: A taxonomy of critical factors. *European Journal of Operational Research, 146*, 352-364.
3. Bajwa, D. S., Garcia, J. E., & Mooney, T. (2004). An integrative framework for the assimilation of enterprise resource planning systems: phases, antecedents, and outcomes. *JOURNAL OF COMPUTER INFORMATION SYSTEMS, 44*, 81-90.
4. Berchet, C., & Habchi, G. (2005). The implementation and deployment of an ERP system: An industrial case study. *56*, 588-605.
5. ERP Implementation Methodologies (2002).
6. Duarte, A. I. M., & Costa, C. J. (2012). *Information systems: life cycle and success.* Paper presented at the Proceedings of the Workshop on Information Systems and Design of Communication.
7. Ehie, I. C., & Madsen, M. (2005). Identifying critical issues in enterprise resource planning (ERP) implementation. *Computers in Industry, 56*, 545-557.
8. Analysis of critical success factors relevance along SAP implementation phases 1019-1025 (2001).
9. Francalanci, C., Iar, C. H., & Lanci, A. F. (2001). Predicting the implementation effort of ERP projects: empirical evidence on SAP/R3. *Journal of Information Technology, 16*, 33-48. doi: 10.1080/02683960010035943
10. Hallikainen, P., Kimpimäki, H., & Kivijärvi, H. (2006). Supporting the Module Sequencing Decision in the ERP Implementation Process. *Proceedings of the 39th Hawaii International Conference on System Sciences - 2006*, 1-10.
11. Hossain, M. M., & Jahed, M. A. (2010). Factors influencing ont the adjustment of ERP system during implementation. *Global Management Review, 4*(3).
12. Kawalek, P., & Wood-Harper, T. (2002). The finding of thorns: user participation in enterprise system implementation. *SIGMIS Database, 33*, 13-22. doi: 10.1145/504350.504355
13. Kumar, V., Maheshwari, B., & Kumar, U. (2003). An investigation of critical management issues in ERP implementation: empirical evidence from Canadian organizations. *Technovation, 23*, 793-807. doi: 10.1016/S0166-4972(02)00015-9
14. Management of Risks in an ERP Implementation Project 20 (Turku School of Economics 2001).
15. Loh, T. C., & Koh, S. C. L. (2004). Critical elements for a successful enterprise resource planning implementation in small-and medium-sized enterprises. *International Journal of Production Research, 42*, 3433-3455.

16. Mabert, V. A., Soni, A., & Venkataramanan, M. A. (2006). Model based interpretation of survey data: A case study of enterprise resource planning implementations. *Mathematical and Computer Modelling, 44*, 16-29. doi: 10.1016/j.mcm.2005.12.008
17. Maguire, S., Ojiako, U., & Said, A. (2010). ERP implementation in Omantel: a case study. *Industrial Management & Data Systems, 110*(1), 78-92. doi: 10.1108/02635571011008416
18. Markus, M. L., & Tanis, C. (2003). The Enterprise System Experience - From Adoption to Success. *Pinnaflex Educational Resources*, 173-207.
19. Marnewick, C., & Labuschagne, L. (2005). A conceptual model for enterprise resource planning (ERP). *Information Management & Computer Security, 13*, 144-155. doi: 10.1108/09685220510589325
20. Momoh, A., Roy, R., & Shehab, E. (2008). A work breakdown structure for implementing and costing an ERP project. *Communications of the IBIMA, 6*(15), 94-103.
21. Ng, J. K. C., Ip, W. H., & Lee, T. C. (1998). The development of an enterprise resources planning system using a hierarchical design pyramid. *Journal of Intelligent Manufacturing, 9*, 385-399. doi: 10.1023/a:1026409806874
22. Parr, A., & Shanks, G. (2000). A model of ERP project implementation. *Journal of Information Technology, 15*, 289-303.
23. Rajagopal, P. (2002). An innovation--diffusion view of implementation of enterprise resource planning (ERP) systems and development of a research model. *Information & management, 40*, 87-114.
24. Rosa, W., Packard, T., Krupanand, A., Bilbro, J. W., & Hodal, M. M. (2013). COTS Integration and Estimation for ERP. *Journal of Systems and Software, 86*, 538-550. doi: 10.1016/j.jss.2012.09.030
25. Şahin, M., & Hamsioglu, D. P. (2008). Enterprise Resource Planning: Comparison Implemantation Procedures of Two Companies. *Lex ET Scientia International Journal (LESIJ), 15*(2), 215-224.
26. Sarker, S., & Lee, A. S. (2003). Using a case study to test the role of three key social enablers in ERP implementation. *Information & management, 40*, 813-829.
27. Using Stakeholder Theory to Analyze Knowledge Sharing During Enterprise Systems Implementations Paper 26 (2004).
28. SCORŢA, I. (2007). Key players and activities across the ERP life cycle: A temporal perspective.
29. Somers, T. M., & Nelson, K. G. (2004). A taxonomy of players and activities across the ERP project life cycle. *Information & management, 41*, 257-278.
30. Subramanian, G. H., & Peslak, A. R. (2010). User perception differences in enterprise resource planning implementations. *JOURNAL OF COMPUTER INFORMATION SYSTEMS, 50*(3), 130-138.
31. Sumner, M. (2000). Risk factors in enterprise-wide/ERP projects. *Journal of Information Technology, 15*, 317-327. doi: 10.1080/02683960010009079

32. Tambovcevs, A. (2012). ERP system implementation in Latvian manufacturing and construction company. *Technological and Economic Development of Economy, 18*(1), 67-83.
33. Tchokogué, A., Bareil, C. C., & Duguay, C. R. (2005). Key lessons from the implementation of an ERP at Pratt & Whitney Canada. *International Journal of Production Economics, 95*, 151-163. doi: 10.1016/j.ijpe.2003.11.013
34. Tsai, W.-H., Hwang, E. T., Chang, J.-C., Lai, C.-W., Lin, S.-J., & Yang, C.-H. (2012). Taxonomy of cost of quality (COQ) across the enterprise resource planning (ERP) implementation phases. *African Journal of Business Management, 6*(3), 854.
35. Umble, E. J., Haft, R. R., & Umble, M. M. (2003). Enterprise resource planning: Implementation procedures and critical success factors. *European Journal of Operational Research, 146*, 241-257.
36. Vathanophas, V. (2007). Business process approach towards an inter-organizational enterprise system. *Business Process Management Journal, 13*, 433-450. doi: 10.1108/14637150710752335
37. An analysis of the imagine PA public sector ERP project 8 (2004).
38. Wei, C.-C., Chien, C.-F., & Wang, M.-J. J. (2005). An AHP-based approach to ERP system selection. *International Journal of Production Economics, 96*, 47-62.
39. Wei, C.-C., & Wang, M.-J. J. (2004). A comprehensive framework for selecting an ERP system. *International Journal of Project Management, 22*, 161-169.
40. Weston, F. C. (2001). ERP implementation and project management. *Production and Inventory Management Journal, 42*, 75-80.
41. Yusuf, Y., Gunasekaran, A., & Abthorpe, M. S. (2004). Enterprise information systems project implementation: A case study of ERP in Rolls-Royce. *International Journal of Production Economics, 87*, 251-266.
42. Zhang, L., & Li, Y. C. (2006). Theory and practice of systems methodology in ERP implementation. *Systems Research and Behavioral Science, 23*, 219-235. doi: 10.1002/sres.754

Appendices Chapter 5

Appendix 5.1 Interview manual step 3

Het interview dient om potentiële problemen van type C en D te inventariseren.

Het interview bestaat daarom uit 3 delen, waarbij er een opbouw is van brainstorming, naar details vragen, naar verificatie van al vastgelegde problemen.

Deel 1: Problemen kort inventariseren

Eerst vragen naar het type problemen dat er gezocht wordt. Daarbij gaat het om de problemen die direct bij de geïnterviewde in gedachten komen als hij/zij denkt aan het ERP project. Het uitgangspunt hierbij is dat problemen die direct naar boven komen, problemen zijn die moeilijk waren en/of waarvoor het lang geduurd heeft voordat er een oplossing was.
Indien er spontaan geen problemen bij de geïnterviewde opkomen, wordt deze geholpen met een voorbeeld van de problemen die uit de documentatie gehaald zijn. In dit deel gaat het erom een lijst van problemen te krijgen die in de rest van het interview verder uitgewerkt kunnen worden.

1. *Wat komt meteen aan lastige, moeilijk oplosbare en/of terugkerende problemen bij u op, als u aan het project denkt?*
2. *Vanuit de stuurgroepverslagen heb ik zelf geprobeerd een lijst op te stellen van dergelijke problemen. Hier is een voorbeeld van een dergelijk probleem. Wellicht dat dit soortgelijke problemen bij u in herinnering brengt.*

Deel 2: Per probleem meer details vastleggen

Per genoemd probleem wordt nu meer naar detail gevraagd. Hierbij wordt ervan uitgegaan dat door het aflopen van de in deel 1 opgestelde lijst er systematischer en duidelijker afgegrensd in de tijd per probleem een toelichting gegeven kan worden, dan als dit direct bij het noemen van het probleem zou gebeuren.
Er wordt gevraagd naar wat het probleem precies inhield, zodat dit duidelijk onderscheiden kan worden van andere problemen en bovendien er voor volgende interviews een goede beschrijving gemaakt kan worden voor verificatie door de volgende geïnterviewden. Om al een idee te krijgen over of het een C- of D-kandidaat is, wordt er gevraagd naar frequentie van vergaderingen en naar het resultaat. Er wordt bewust niet gevraagd naar door wie en wat over het probleem besloten is, om hiermee te voorkomen dat het onderzoeksdoel bekend wordt.

1. *Wilt u mij in het kort uitleggen wat het probleem precies inhield?*
2. *Hoe vaak is er over dit probleem vergaderd binnen de stuurgroep?*
3. *Wat was het resultaat?*
4. *Heeft het al dan niet aanpakken van dit probleem nog andere problemen? Indien ja, welke problemen?*

Deel 3: Het verifiëren van de al aangelegde lijst van problemen

In het laatste onderdeel wordt de lijst die al is opgesteld door documentonderzoek en aangevuld is met vorige interviews doorgelopen en geverifieerd. Uiteraard indien hierbij nieuwe problemen in gedachten komen, worden deze weer in detail besproken.
Dit deel van het interview dient meteen als triangulatie van het documentonderzoek en ook van de resultaten van de vorige interviews. Daarnaast kan het het ook nog stimuleren tot het noemen van nieuwe problemen. Dit wordt bewust als laatste onderdeel uitgevoerd om te voorkomen dat je al opgestelde lijst het brainstormen van de geïnterviewde over problemen zou beïnvloeden.

1. *Is dit probleem al door u genoemd?*
2. *Indien het probleem nog niet door u genoemd is, ziet u het als een type probleem waarnaar ik op zoek ben? Waarom wel/niet?*

Interview protocol voor stap 3
In dit onderdeel wordt de structuur van het interview besproken. Uiteraard zal de tekst niet letterlijk opgelezen worden, maar wordt de inhoud al naar gelang de situatie en geïnterviewde geparafraseerd.

Introductie
Hartelijk dank dat ik u mag interviewen.

Introductie mijzelf: Ik ben …… werk bij ….. als …… doe ….

Ik heb u benaderd om aan dit interview deel te nemen omdat u goed op de hoogte bent van de inhoud en gebeurtenissen van het ERP project _____ bij _____.

In mijn wetenschappelijk onderzoek probeer ik meer inzicht te krijgen in de problematiek van ERP projecten. Door het vastleggen van de problemen zoals die zich tijdens een concreet ERP project hebben voorgedaan en deze te vergelijken met problemen binnen andere organisaties, kan ik mogelijkerwijs inzichten verkrijgen waarmee ik richtlijnen kan ontwerpen voor verbetering van dergelijke trajecten, met andere woorden de complexiteit wat kan verminderen.

In ons interview ligt daarom de nadruk op het inventariseren van problemen die gedurende het ERP project hebben gespeeld die gekenmerkt worden doordat ze lastig waren op te lossen en/of vaker terug kwamen. Uiteraard gaat het alleen om de problemen die voor u zichtbaar waren of waarvan u voldoende op de hoogte was.

Vertrouwelijkheid informatie
Om het voor mij gemakkelijker te maken om de informatie zo objectief mogelijk te kunnen verwerken, wil ik graag ons gesprek opnemen met deze memorecorder. Op die manier kan ik alle details vastleggen en toch tegelijkertijd aandacht hebben voor ons gesprek.

Uiteraard zijn de opname, de aantekeningen en uitwerking van dit interview volledig vertrouwelijk en wordt er behalve in overleg met medeonderzoekers (die ook gehouden zijn aan strikte geheimhoudingsvoorwaarden) hiervan niets publiek gemaakt, noch binnen uw organisatie, noch buiten uw organisatie.

Alleen de algemene conclusies die uit de totale serie van interviews getrokken kunnen worden, zullen gepubliceerd worden en binnen uw organisatie gepubliceerd.

In deze publicaties zal er op geen enkele wijze verwezen worden naar u, of zullen gegevens (bijvoorbeeld uitspraken) worden opgenomen die op enigerlei wijze naar u herleid kunnen worden.
Indien u dit wenst zal ik hiervoor nog een ondertekende verklaring opstellen.

Uw deelname aan dit interview is uiteraard vrijwillig en u kunt op elk moment het interview beëindigen indien u zich ongemakkelijk voelt. Het is niet de bedoeling dat dit interview u op welke wijze dan ook schade berokkent of vervelend is.

Administratie interview
Voordat wij beginnen zou ik graag eerst even wat administratieve gegevens willen vastleggen of laten controleren.

Interviewer vult de identificatiegegevens van het interview zoveel mogelijk vooraf in en vraagt aan het begin aan de geïnterviewde indien nodig nog missende gegevens en controleert samen met de geïnterviewde de gegevens.

Naam geïnterviewde	
Man/Vrouw	
Telefoonnummer geïnterviewde	
E-mail geïnterviewde	
Afdeling geïnterviewde	
Organisatie geïnterviewde	
Datum interview	
Tijdstip interview	
Plaats interview	
Duur interview	
Naam interviewer	

Werkwijze interview
Mag ik u al op voorhand bedanken voor uw bereidheid om aan dit interview mee te werken.

Het interview zal niet langer dan 1,5 uur duren. Bij behoefte zullen wij een korte pauze inlassen.

Ik zal u verschillende open vragen stellen over de problemen tijdens het ERP project. Indien nodig zal ik op details doorvragen.

Echter u bepaalt natuurlijk zelf óf, en in welke mate u de vragen wilt beantwoorden.

Ik nodig u alleen uit om te vertellen wat u weet over de onderwerpen, er zijn geen goede of foute antwoorden.

Geef ook gerust aan wanneer u een vraag niet kunt of wilt beantwoorden. U hoeft mij geen reden hiervoor te geven.

Persoonlijke gegevens
Ik zou u graag eerst een aantal vragen willen stellen over uw functie en achtergrond:

1. Wat is uw huidige functie?
2. Indien anders, wat was uw functie tijdens het ERP project?
3. Wat was uw rol/rollen tijdens het ERP project en gedurende welke periode of perioden?
4. Hoeveel jaar ervaring hebt u in uw huidige functie?
5. Hoeveel jaar ervaring hebt u met ERP implementatieprojecten?

Hoofdinterviewvragen
Zoals al in mijn e-mail genoemd, wil ik middels dit interview problemen van dit ERP project vastleggen die lastig waren op te lossen en/of misschien vaker weer naar voren kwamen. In mijn onderzoek probeer ik aan de hand van deze problemen en soortgelijke problemen uit andere organisaties meer inzicht te krijgen in de complexiteit van ERP implementaties en vanuit dat inzicht bij te dragen aan het verminderen hiervan.

In dit eerste deel van mijn onderzoek wil ik daarom eerst inventariseren welke problemen van dit type er in het project zijn opgetreden.
Dus in ons interview gaat het er mij vooral om, om de problemen te detecteren en te begrijpen. Het gaat nog niet om diepgaand de reden waarom een probleem is opgetreden te bespreken en de historie hoe het probleem uiteindelijk wel/niet is opgelost. Dit komt in een latere fase aan de orde.

Problemen kort inventariseren
Eerst zou ik graag met u een lijst maken van de volgens u belangrijkste lastige problemen die zich tijdens het ERP project hebben voorgedaan.
Daarna wil ik graag aan de hand van enkele vragen iets dieper op ieder van deze problemen ingaan om de essentie te begrijpen.

1. Wat komt meteen aan lastige, moeilijk oplosbare en/of terugkerende problemen bij u op, als u aan het project denkt?

(De volgende vraag stellen indien niet direct of onvoldoende bij de geïnterviewde een lijstje met dit type problemen naar voren komt)
2. Vanuit de stuurgroepverslagen heb ik zelf geprobeerd een lijst op te stellen van dergelijke problemen. Hier is een voorbeeld van een dergelijk probleem. Wellicht dat dit soortgelijke problemen bij u in herinnering brengt.

Per probleem meer details vastleggen
Nu zou ik aan de hand van enkele vragen ieder probleem iets verder in detail willen bespreken.

1. **Wilt u mij in het kort uitleggen wat het probleem precies inhield?**

2. **Hoe vaak is er over dit probleem vergaderd binnen de stuurgroep?**

3. **Wat was het resultaat?**

4. **Heeft het al dan niet aanpakken van dit probleem nog andere problemen veroorzaakt? Indien ja, welke problemen?**

Het verifiëren van de al aangelegde lijst van problemen
(Interviewer geeft lijst van problemen uit stuurgroep verslagen en aanvullingen vanuit vorige interviews aan geïnterviewde en licht deze toe. Dit dient ter verificatie van de lijst en indien er bij de geïnterviewde geen problemen waren opgekomen eventueel tot het herinneren van andere problemen.)
Uit de project-documenten (en uit vorige interviews) is de volgende lijst opgesteld van problemen……
Per probleem wil ik graag van u het volgende weten:

1. **Is dit probleem al door u genoemd?**

2. **Indien het probleem nog niet door u genoemd is, ziet u het als een type probleem waarnaar ik op zoek ben? Waarom wel/niet?**

Afsluiting interview
Ik denk dat ik hiermee voldoende informatie gekregen heb.
De audio-opname en aantekeningen worden door mij na ons interview verder verwerkt.

Ik zal de door u genoemde problemen tijdens het ERP project steeds kort beschrijvingen en deze in de vorm van een kort verslag naar u toesturen per e-mail voor verificatie.
Ik zal dit binnen enkele dagen doen.

Ik zal na 1 week telefonisch contact met u opnemen om te vragen of u nog opmerkingen, aanmerkingen of aanvullingen hebt op dit verslag.

Als ik die heb aangebracht, ontvangt u de definitieve versie waarvoor ik graag van u, als u het eens bent met de inhoud, een korte bevestiging (per email of telefonisch) zou willen ontvangen over de acceptatie van de inhoud.

Mag ik u van harte danken voor de medewerking aan dit interview en dus ook uw medewerking aan een stukje wetenschappelijk onderzoek. Zonder medewerking zoals de uwe met praktijk-ervaringen, kunnen wij op dit gebied geen wetenschappelijk onderzoek doen en daardoor nuttige bijdragen leveren aan, in dit geval, toekomstige ERP projecten.

Uiteraard wordt u aan het einde van dit onderzoek persoonlijk uitgenodigd voor een presentatie die ik zal houden binnen uw organisatie over de resultaten.

Afsluiting administratie interview
Na interview invullen:

Duur interview	
Naam bestand audioopname	

Benodigde hulpmiddelen, faciliteiten en achtergrond informatie voor stap 3
Faciliteiten per interview:

- Overlegkamer voor 2 personen, bij voorkeur het kantoor van de geïnterviewde

Hulpmiddelen interviewer:

- Memorecorder
- Horloge
- Pen en papier
- Vragenlijst
- Lijst met uit projectdocumentatie/stuurgroep verslagen afgeleide problemen en aanvullingen uit eventuele vorige interviews

Standaard E-mails voor stap 3
In dit onderdeel zijn de sjablonen voor de e-mails weergegeven om het interview proces te begeleiden.

Uitnodiging

Verzoek voor een interview vanuit de Open Universiteit
_____

Geachte _____.

Ik ben docent bij de Open Universiteit bij de faculteit Management, Science & Technology.

Van _____ heb ik een bevestiging ontvangen dat u bereid bent om deel te nemen aan een interview en daarmee bij te dragen aan een wetenschappelijk onderzoek op het gebied van de problematiek van ERP projecten. Hiervoor alvast hartelijk dank.

Het doel van mijn wetenschappelijk onderzoek is om door meer inzicht te verkrijgen in reële ERP projecten, betere inzichten te krijgen in de problemen van ERP projecten in zijn algemeenheid. Op deze wijze hoop ik uiteindelijk eraan bij te dragen dat ERP projecten in de praktijk beter gemanaged kunnen worden.

Daarom wil ik graag van het ERP project _____ van _____ de opgetreden problemen via interviews in kaart brengen.

Graag zou ik u daarom willen interviewen, waarbij wij aan de hand van enkele open vragen de belangrijkste problemen die u zich herinnert van het ERP project _____ inventariseren.

Het interview zal plaatsvinden op een locatie naar uw keuze en zal **maximaal 1,5 uur** van uw tijd in beslag nemen (naar verwachting minder).

Als enige voorbereiding voor dit interview zou ik u willen vragen om van tevoren na te denken over wat volgens u de belangrijkste lastige problemen waren die zich tijdens het project hebben voorgedaan of wellicht nog steeds aanwezig zijn. Deze problemen kunnen op allerlei gebied spelen zoals organisatorisch, procesmatig, technisch et cetera. Het type probleem dat ik probeer vast te leggen werd/wordt binnen het pro-

ject als een lastig probleem gezien, waarvoor niet direct een oplossing voor bedacht kon worden of een probleem dat steeds weer de kop opstak.

Na het interview zal ik de resultaten uitwerken en u een kort verslag sturen ter verificatie.

Uiteraard zal ik de resultaten uit het complete onderzoek bij _____ komen presenteren.

Als u hier geen bezwaar tegen hebt, zou ik graag in de komende dagen telefonisch contact met u willen opnemen om een afspraak voor het interview te maken. In dat telefonisch contact kan ik u ook nog verdere toelichting op het interview geven.

Met vriendelijke groet,

Guy Janssens

Bevestiging interview afspraak
Geachte _____,

Hierbij bevestig ik onze afspraak voor ons interview op _____ te _____ van __ tot __ uur.

Met vriendelijke groet,

Guy Janssens

Begeleiding uitwerking interview
Uitwerking interview ter verificatie en aanvulling
_____

Geachte _____,

Nogmaals hartelijk dank voor ons interview op _____ te _____.

Zoals afgesproken ontvangt u hierbij mijn uitwerking van wat er in het interview besproken is.
Het is een lijst van problemen die tijdens het project zijn opgetreden met een korte beschrijving van de inhoud .

Graag verzoek ik u te beoordelen of ik de problemen correct en volledig heb weergegeven.
Mochten er correcties en/of aanvullingen nodig zijn dan kunt u die mij dat via mail aangeven of indien u dit prettiger vind, in een telefonisch of face-to-face gesprek.

Ik neem in ieder geval binnen 1 week contact met u hierover op.

Met vriendelijke groet,

Guy Janssens

Appendix 5.2 Interview manual step 6

Voor het kunnen classificeren van ieder probleem in A, B, C en D is het nodig om voldoende informatie te hebben over de inhoud van ieder probleem.

Hiervoor zijn de volgende opties mogelijk:

1. Alle potentiële problemen uitwerken als minicases, middels zeer gedetailleerde vragen op basis van het "issue and event handling" model. Hierna aan de hand van de rijke informatie die per probleem bekend is de classificatie uitvoeren.

Voordelen:

- De classificatie is voor andere onderzoekers volledig transparant.
- Voor het onderzoek is dit de meest rijke uitwerking aan gegevens.
- Wat er precies in het onderzoek wordt onderzocht, is voor de geïnterviewde minder zichtbaar.

Nadelen:

- Elke minicase kost veel tijd in interviewtijd, doorlooptijd en uitwerking
- Er wordt veel tijd gevraagd van de geïnterviewde zowel tijdens het interview als bij de verificatie van de uitwerking, waardoor bereidheid tot deelname minder kan zijn.
- De geïnterviewde zal ook reageren op aspecten die niet van belang zijn voor het onderzoek, waardoor het voor de interviewer moeilijker is om het interview in het juiste spoor te houden.

2. Door middel van vragen die alleen te vragen naar de beslisser en verantwoordelijkheid, de richting van het empirisch onderzoek onthullen en zo op een efficiënte en effectieve wijze aan de hand van de antwoorden de potentiële problemen kunnen classificeren.

Voordelen:

- Er wordt direct doorgestoten naar de kern van wat het onderzoek wil weten om conclusies te kunnen trekken. Dit wil zeggen rechtstreeks te vragen naar wie een beslissing heeft genomen en binnen wiens verantwoordelijkheden dit viel, kan de classificatie objectief gemaakt worden.
- Hierdoor wordt met minimale inspanning toch een verantwoorde classificatie bereikt.

Nadelen:

- Door duidelijk te maken wat het interview precies wil weten, kan het zijn dat de medewerking vermindert, ten gevolgde van gevoeligheden op het gebied van verantwoordelijkheden.
- Het eindresultaat laat alleen de aspecten beslisser en verantwoordelijkheid zien en de overige beschrijvingen van het probleem (besluitproces, acties e.d.) wordt niet vastgelegd.

3. De organisatie/geïnterviewden het model uitleggen en ze dan zelf aan de hand van dit model de lijst van potentiele problemen laten classificeren.
Voordelen:

- De meest snelle en efficiënte wijze om een classificatie te krijgen.

Nadelen:

- De vraag is of iedere geïnterviewde de classificatie gelijk interpreteert en dus de inhoud snapt.
- Geïnterviewden kunnen classificeren naar "wenselijkheid" van de organisatie of persoon.

Uit de 3 mogelijkheden wordt optie 2 gekozen, omdat met de huidige kennis de verwachting is dat deze optie objectief en efficiënt de benodigde informatie kan opleveren. Bij optie 3 wegen de nadelen van niet duidelijk zijn van de classificatie en onbewuste sturing door "wenselijkheid" te zwaar. Optie 1 levert de meest rijke informatie op, echter het is de vraag of de wellicht rijkere informatie die gebruikt wordt om de classificatie te kunnen uitvoeren, in verhouding staat tot het risico van afhaken van de geïnterviewden door de grotere inspanning die dit van hen vergt. Optie 2 lijkt daarom het meest geschikt zowel om met de gegevens de classificatie objectief te kunnen uitvoeren alsook de kans op de bereidheid van de geïnterviewden om mee te werken het grootst.

Overigens mocht na de eerste interviews uit stap 3 blijken dat er argumenten zijn om toch ieder probleem als minicase uit te werken, dan zal deze keuze aangepast worden. Ook afhankelijk van de resultaten uit de interviews uit stap 3 zal er minstens 1 probleem als minicase uitgewerkt worden, aangezien dit ook gebruikt kan worden in het proefschrift als illustratie van de typen problemen die onderzocht worden.

In het volgende wordt optie 2 beschreven:

Het interview dient om bij potentiële problemen van type C en D die uit de interviews van stap 3 gekomen zijn in detail te vragen naar de doelstelling van dit empi-

risch onderzoek, namelijk kijken of het binnen of buiten de scope en bevoegdheden van het project viel.

Om dit te kunnen doen, zijn de interview vragen zeer direct geformuleerd en wordt er specifiek gevraagd naar beslissers, beslissingen en formele verantwoordelijkheden. Het interview start met het nog eens laten beschrijven van het probleem, zodat duidelijk is dat het te behandelen probleem ook inderdaad hetzelfde is zoals dit door de interviewer is vastgelegd. Daarna wordt gevraagd naar de beslissingen, oplossingen en formele verantwoordelijkheden. Als duidelijk is wie een beslissing heeft genomen, kan dit later afgezet worden tegen de al bekende formele beslissingsbevoegdheden in het project. Als het probleem niet is opgelost wordt gevraagd naar diegene die dit zou moeten beslissen.
Daarna wordt gevraagd naar de rol van de beslisser, onafhankelijk van of het probleem al opgelost is of nog opgelost moet worden. Hierdoor kan naderhand vergeleken worden of de formele verantwoordelijkheid misschien tijdens het project veranderd is of dat er verantwoordelijkheid was die niet officieel in projectdocumentatie was vastgelegd.
Tenslotte wordt gevraagd of er nog documenten zijn binnen de organisatie die zinvol zijn om in relatie tot dit probleem te bekijken.

1. Kunt u in uw eigen woorden nog eens een korte beschrijving geven van het probleem?
2. Is het probleem opgelost?
- Indien ja: *Wie heeft de beslissing genomen voor de oplossing?*
Wat was de oplossing?

- Indien nee: *Wie was of is volgens u verantwoordelijk voor de oplossing van dit probleem?*

3. Wat was/is de rol in het project van diegene die de beslissing heeft genomen of formeel verantwoordelijk hiervoor was/is?
4. Zijn er volgens u nog documenten die voor mij zinvol zouden zijn om met betrekking tot dit probleem te bestuderen? Indien ja, kunt en/of wilt u mij deze ter beschikking stellen of aangeven hoe kan ik toegang zou kunnen verkrijgen tot die documenten?

Interview protocol voor stap 6
In dit onderdeel wordt de structuur van het interview besproken. Uiteraard zal de tekst niet letterlijk opgelezen worden, maar wordt de inhoud al naar gelang de situatie en geïnterviewde geparafraseerd.

Introductie
Hartelijk dank dat ik u mag interviewen.

Introductie mijzelf: Ik ben …… werk bij ….. als …… doe ….

Ik heb u benaderd om aan dit interview deel te nemen omdat u goed op de hoogte bent van de inhoud van de volgende 2 problemen van het ERP project _____ bij _____.
1._____
2._____

In mijn wetenschappelijk onderzoek probeer ik meer inzicht te krijgen in de problematiek van ERP projecten. Door het vastleggen van de problemen zoals die zich tijdens een concreet ERP project hebben voorgedaan en deze te vergelijken met problemen binnen andere organisaties, kan ik mogelijkerwijs inzichten verkrijgen waarmee ik richtlijnen kan ontwerpen voor verbetering van dergelijke trajecten, met andere woorden de complexiteit wat kan verminderen.

In ons interview ligt daarom de nadruk op wat er zich bij iedere van deze problemen gedurende het ERP project heeft afgespeeld. Uiteraard gaat het alleen om datgene dat voor u zichtbaar was of waarvan u voldoende op de hoogte was.

Vertrouwelijkheid informatie
Om het voor mij gemakkelijker te maken om de informatie zo objectief mogelijk te kunnen verwerken, wil ik graag ons gesprek opnemen met deze memorecorder. Op die manier kan ik alle details vastleggen en toch tegelijkertijd aandacht hebben voor ons gesprek.

Uiteraard zijn de opname, de aantekeningen en uitwerking van dit interview volledig vertrouwelijk en wordt er behalve in overleg met medeonderzoekers (die ook gehouden zijn aan strikte geheimhoudingsvoorwaarden) hiervan niets publiek gemaakt, noch binnen uw organisatie, noch buiten uw organisatie.

Alleen de algemene conclusies die uit de totale serie van interviews getrokken kunnen worden, zullen gepubliceerd worden en binnen uw organisatie gepubliceerd.

In deze publicaties zal er op geen enkele wijze verwezen worden naar u, of zullen gegevens (bijvoorbeeld uitspraken) worden opgenomen die op enigerlei wijze naar u herleid kunnen worden.
Indien u dit wenst zal ik hiervoor nog een ondertekende verklaring opstellen.

Uw deelname aan dit interview is uiteraard vrijwillig en u kunt op elk moment het interview beëindigen indien u zich ongemakkelijk voelt. Het is niet de bedoeling dat dit interview u op welke wijze dan ook schade berokkent of vervelend is.

Administratie interview
Voordat wij beginnen zou ik graag eerst even wat administratieve gegevens willen vastleggen of laten controleren.

Interviewer vult de identificatiegegevens van het interview zoveel mogelijk vooraf in en vraagt aan het begin aan de geïnterviewde indien nodig nog missende gegevens en controleert samen met de geïnterviewde de gegevens.

Naam geïnterviewde	
Man/Vrouw	
Telefoonnummer geïnterviewde	
E-mail geïnterviewde	
Afdeling geïnterviewde	
Organisatie geïnterviewde	
Datum interview	
Tijdstip interview	
Plaats interview	
Duur interview	
Naam interviewer	
Te bespreken probleem 1	
Te bespreken probleem 2	

Werkwijze interview
Mag ik u al op voorhand bedanken voor uw bereidheid om aan dit interview mee te werken.

Het interview zal niet langer dan 1 uur duren. Bij behoefte zullen wij een korte pauze inlassen.

Ik zal u verschillende open vragen stellen over twee problemen tijdens het ERP project.
Indien nodig zal ik op details doorvragen.

Echter u bepaalt natuurlijk zelf óf, en in welke mate u de vragen wilt beantwoorden.

Ik nodig u alleen uit om te vertellen wat u weet over de onderwerpen, er zijn geen goede of foute antwoorden.

Geef ook gerust aan wanneer u een vraag niet kunt of wilt beantwoorden. U hoeft mij geen reden hiervoor te geven.

Persoonlijke gegevens
Ik zou u graag eerst een aantal vragen willen stellen over uw functie en achtergrond:

1. Wat is uw huidige functie?
2. Indien anders, wat was uw functie tijdens het ERP project?
3. Wat was uw rol/rollen tijdens het ERP project en gedurende welke periode of perioden?
4. Hoeveel jaar ervaring hebt u in uw huidige functie?
5. Hoeveel jaar ervaring hebt u met ERP implementatieprojecten?

Hoofdinterviewvragen
Via de volgende vragen wil ik graag uw visie krijgen op de volgende 2 problemen die tijdens het ERP project hebben gespeeld.

Interviewer geeft per probleem een kleine inleiding met in het kort de samengevatte informatie uit de eerdere interviews.

Vragen per probleem:

1. Kunt u in uw eigen woorden nog eens een korte beschrijving geven van het probleem?

2. Is het probleem opgelost?

- Indien ja: wie heeft de beslissing genomen voor de oplossing?
- Wat was de oplossing?

- Indien nee: wie was of is volgens u verantwoordelijk voor oplossing van dit probleem?

3. Wat was/is de rol in het project van diegene die de beslissing heeft genomen of formeel verantwoordelijk hiervoor was/is?

4. Zijn er volgens u nog documenten die voor mij zinvol zouden zijn om met betrekking tot dit probleem te bestuderen? Indien ja, kunt en/of wilt u mij deze ter beschikking stellen of aangeven hoe kan ik toegang zou kunnen verkrijgen tot die documenten?

Interviewer checklist voor probleem:

- *Duidelijkheid over wat precies het probleem was.*
- *Opgelost J/N (recurring J/N)*
- *Oplossing zelf*
- *Personen betrokken bij probleem*
- *Rol van personen*
- *Genomen beslissingen*
- *Formele verantwoordelijkheden*
- *Acties genomen voor oplossing*
- *Tijdpad*
- *Aanvullende documentatie*

Afsluiting interview
Ik denk dat ik hiermee voldoende informatie gekregen heb.
De audio-opname en aantekeningen worden door mij na ons interview verder verwerkt.

Ik zal de door u genoemde informatie per probleem opschrijven en deze in de vorm van een kort verslag naar u toesturen per e-mail voor verificatie.
Ik zal dit binnen 1 week doen.

Ik zal na 2 weken telefonisch contact met u opnemen om te vragen of u nog opmerkingen, aanmerkingen of aanvullingen hebt op dit verslag.

Als ik die heb aangebracht, ontvangt u de definitieve versie waarvoor ik graag van u, als u het eens bent met de inhoud, een korte bevestiging (per email of telefonisch) zou willen ontvangen over de acceptatie van de inhoud.

Mag ik u van harte danken voor de medewerking aan dit interview en dus ook uw medewerking aan een stukje wetenschappelijk onderzoek. Zonder medewerking zoals de uwe met praktijkervaringen, kunnen wij op dit gebied geen wetenschappelijk onderzoek doen en daardoor nuttige bijdragen leveren aan in dit geval toekomstige ERP projecten.

Uiteraard wordt u aan het einde van dit onderzoek persoonlijk uitgenodigd voor een presentatie die ik zal houden binnen uw organisatie over de resultaten.

Afsluiting administratie interview
Na interview invullen:

Duur interview	
Naam bestand audioopname	
Opsomming extra documenten voor bestuderen en weg om deze te verkrijgen.	

Benodigde hulpmiddelen, faciliteiten en achtergrond informatie voor stap 6
Faciliteiten per interview:

- Overlegkamer voor 2 personen, bij voorkeur het kantoor van de geïnterviewde

Hulpmiddelen interviewer:

- Memorecorder
- Horloge
- Pen en papier
- Vragenlijst
- Het verslag van de twee te bespreken problemen

Standaard E-mails voor stap 6
In dit onderdeel zijn de sjablonen voor de e-mails weergegeven om het interview proces te begeleiden.

Uitnodiging

Verzoek voor een interview vanuit de Open Universiteit
_____

Geachte _____.

Ik ben docent bij de Open Universiteit bij de faculteit Management, Science & Technology.

Van _____ heb ik een bevestiging ontvangen dat u bereid bent om deel te nemen aan een interview en daarmee bij te dragen aan een wetenschappelijk onderzoek op het gebied van de complexiteit van ERP projecten. Hiervoor alvast hartelijk dank.

Het doel van mijn wetenschappelijk onderzoek is om door meer inzicht te verkrijgen in reële ERP projecten, betere inzichten te krijgen in de problemen van ERP projecten in zijn algemeenheid. Op deze wijze hoop ik uiteindelijk eraan bij te dragen dat ERP projecten in de praktijk beter gemanaged kunnen worden.

Daarom wil ik graag van het ERP project _____ van _____ De volgende twee opgetreden problemen via ons interview in kaart brengen:
1. _____
2. _____

Graag zou ik u daarom willen interviewen, waarbij wij aan de hand van enkele open vragen deze twee problemen van het ERP project _____ bespreken.

Het interview zal plaatsvinden op een locatie naar uw keuze en zal **maximaal 1 uur** van uw tijd in beslag nemen.

Als enige voorbereiding voor dit interview zou ik u willen vragen om van tevoren na te denken over er precies aan de hand was bij de bovengenoemde problemen.

Na het interview zal ik de resultaten uitwerken en u een kort verslag sturen ter verificatie.

Uiteraard zal ik de resultaten uit het complete onderzoek bij _____ komen presenteren.

Als u hier geen bezwaar tegen hebt, zou ik graag in de komende dagen telefonisch contact met u willen opnemen om een afspraak voor het interview te maken. In dat telefonisch contact kan ik u ook nog verdere toelichting op het interview geven.

Met vriendelijke groet,

Guy Janssens

Bevestiging interview afspraak
Geachte _____,

Hierbij bevestig ik onze afspraak voor ons interview op _____ te _____ van __ tot __ uur.

Met vriendelijke groet,

Guy Janssens

Begeleiding uitwerking interview
Uitwerking interview ter verificatie en aanvulling
_____

Geachte _____,

Nogmaals hartelijk dank voor ons interview op _____ te _____.

Zoals afgesproken ontvangt u hierbij mijn uitwerking van wat er in het interview besproken is.

Graag verzoek ik u te beoordelen of ik de twee problemen correct en volledig heb weergegeven.
Mochten er correcties en/of aanvullingen nodig zijn dan kunt u die mij dat via mail aangeven of indien u dit prettiger vind, in een telefonisch of face-to-face gesprek.

Ik neem in ieder geval binnen 1 week contact met u hierover op.

Met vriendelijke groet,

Guy Janssens

Appendix 5.3 Comments from case organization during presentation of research results

1. Het kan natuurlijk ook zijn dat er positieve onverwachte gebeurtenissen optreden. Nu wordt bij het onderzoek alleen op de negatieve dingen gelet. Moet je hierbij dan dus niets doen? Sommigen zien dit als een afwijking van het oorspronkelijke plan en dan is dat dus ook negatief. Vreemd toch?
2. Er heerst een sterke technologiefocus bij ERP projecten. IT is in de lead.
3. De projectmanager heeft geen ruimte om de projectkaders ter discussie te stellen. Hij wordt afgerekend op de gemaakte afspraken en zal ook uitkijken om dat te doen.
4. Aanwezigen benadrukken ook het verwachtingsmanagement bij (toekomstige)-ERP gebruikers. Bij projectmanagers is het wel duidelijk dat de organisatie centraal staat en dat je aan de verwachtingen van gebruikers tegemoet moet komen, en toch wordt er gedaan alsof je een simpele technische ingreep doet. Ondank dat de men zich bewust is van het feit dat bij een ERP implementatie de organisatie centraal staat, wordt het project toch als IT-project beschouwd en aangestuurd.
5. Algehele erkenning van de opmerkingen en conclusie van de onderzoeker
   "Dat onverwachte problemen voorkomen bij ERP implementaties zien we bij onze organisatie inderdaad terug."
   "Het derde type complexiteit is bij onze organisatie bij ERP implementaties te herkennen."
   "Tja, we voeden al vele millennia kinderen op en tot op heden hebben we kennelijk niet geleerd hoe dit foutloos te doen. Er is kennelijk iets wat altijd onbekend en onverwacht blijft?"
6. Typisch bij onze organisatie dat iedereen zich met alles kan bemoeien, terwijl dit toch vreemd zou moeten zijn. Er zijn immers formele functies en verantwoordelijkheden vastgesteld, ook in een ERP project. Zo ontstaat onduidelijkheid over de legitimiteit van sommige stuuracties.
7. Het is misschien wel vreemd om dit te zeggen, maar onze organisatie heeft geen echte geldnood en kan zich van alles permitteren en dus zal er niet zoveel veranderen.
8. Het derde type complexiteit en de bijbehorende onverwachte problemen gelden ook voor andere grote projecten die invloed hebben op het functioneren van de organisatie.