ANGLIA RUSKIN UNIVERSITY
CAMBRIDGE

A FRAMEWORK FOR MODELLING COMPLEX BUSINESS SYSTEMS FOR TRANSITION TO CLOUD HOSTED SERVICES

KENNETH JOSEPH SPITERI

A thesis in partial fulfilment of the
requirements of Anglia Ruskin University
for the degree of
Doctor of Philosophy

Acknowledgements

This dissertation was prepared in part fulfilment of the requirements for the degree of Doctor of Philosophy. I would like to express my sincere gratitude to my supervisors Tim Reynolds, Dr. Cristina Luca and Dr. George Wilson at Anglia Ruskin University, for their guidance and encouragement. Without their help, this dissertation would not have been possible.

Thanks also to Dr. Aled Jones for his support at my annual review meetings, Dr. Arooj Fatima for her general advice, and Catherine Wijman for proof-reading.

Grateful to Prof. Sergiu Dascalu of the University of Nevada and Dr. Mike Hobbs of Anglia Ruskin University for their willingness to examine the degree.

Last, but certainly not least, I am indebted to my family; my parents for their continuous support and life-long encouragement, and my wife Emmanuela who lovingly stood by me throughout it all. Special appreciation goes to my daughter Alicia, whose unknowing smiles provided joyful distraction, and to my son Daniel, whose arrival coincided with the completion of this journey.

The research work disclosed in this publication is partially funded by the Malta Government Scholarship Scheme grant.

Dedication

Dedicated to my parents Alfred and Mary

ANGLIA RUSKIN UNIVERSITY

ABSTRACT

FACULTY OF SCIENCE AND TECHNOLOGY

DOCTOR OF PHILOSOPHY

A FRAMEWORK FOR MODELLING COMPLEX BUSINESS SYSTEMS FOR TRANSITION TO CLOUD HOSTED SERVICES

KENNETH J. SPITERI

Complexity within business and its supporting systems is interlinked with the evolution of Cloud computing and Enterprise Resource Planning (ERP) systems. Business complexity is therefore a key factor in strategic technology investments such as a transition to Cloud hosted Software-as-a-Service (SaaS) solutions, as this has an impact on risk, cost and requirements. It is observed in practice however, that complexity is primarily a qualitative consideration within the early stages of current ERP implementations. Within a business context, existing validated complexity measures include Control Flow Complexity (CFC) and Redundancy Measure, which evaluate a business at the process level, but assume a static scenario inappropriate for a system implementation that can change through different configuration permutations. Previous studies have attempted to compare various complexity measures to identify a common element but have failed to establish a unique derivative measure, with comparison outcomes being reported in qualitative attributes such as reliability and ease of use. Furthermore, most of these metrics have not been validated theoretically or empirically, and associated tools for their application have not been made publicly available.

The current research has identified a need to derive an alternative definition of business complexity and to develop a framework to analyse and quantify this property as a single measure. The proposed Spiteri Complexity Metric (SCM) developed for this research is a complexity measure that is process and system independent, and utilises a framework to quantify the complexity at an earlier point within the strategic ERP decision process compared to existing approaches. The SCM was theoretically validated using existing robust methods, whilst empirical validation was undertaken utilising 9 business case studies covering 32 companies. An index of complexity for each business case study enabled the calibration of the various components of the framework and required quantitative evaluation of approximately 700 million records, 48,000 data entities and 65 data mining and analyses worksheets.

This work provides a validated complexity toolset that enables combining, unifying and connecting different complexity dimensions across business processes and systems architectures. A comparative calculation of business complexity using the Spiteri

Complexity Framework (SCF) has a myriad of applications. Within a Cloud transition process this includes the quantification of risk, budget forecasts, SaaS service applicability, industry analyses, requirements analyses, service and system comparison, and time estimations. The possibilities for further developing the work to include scaling and extending automation tools for easier application and analyses are discussed.

Keywords: *Complex Systems, Business Systems, Business Complexity, Cloud Computing, Framework Modelling, Enterprise Resource Planning Systems (ERP), Software-as-a-Service (SaaS), Complexity Metric, Complexity Index, Unified Model.*

TABLE OF CONTENTS

LIST OF FIGURES

List of Tables

A FRAMEWORK FOR MODELLING COMPLEX BUSINESS SYSTEMS FOR TRANSITION TO CLOUD HOSTED SERVICES

KENNETH JOSEPH SPITERI

CHAPTER 1

INTRODUCTION

1.1 Background

This chapter examines the commercial landscape that managed IT business services operate in, and highlights how their importance in growth for business served as a motivation for undertaking this work. Analysis of the problems faced by complex business when considering a Software-as-a-Service (SaaS) solution, together with the global growth forecasts of the cloud services industry, helped formulate the primary research question. This led to the definition of the aims and objectives. The research structure is outlined, and serves to frame the context of this thesis, together with the business drivers that push for the development and implementation of toolsets, such as the novel framework developed here-in.

1.2 Research Context and Importance

Businesses have been deploying technology to meet the challenges of embracing the global knowledge economy, and this has been achieved by many through Enterprise Resource Planning (ERP) software. With the advent of cloud services, a new paradigm for these business systems is proposed.

The cloud model suggests that services could be managed by hosting ERP solution providers, with businesses requiring only internet enabled devices to access these services. Amrhein (2009) and others suggest that innovation in distributed computing and access to high-speed internet has increased the feasibility and interest in SaaS, particularly for enterprise services.

Detailed further within the market drivers section, the growth being experienced by both ERP solutions and hosted business services highlights the importance of investigating the impact of one technology model over the other. The technology innovation being made in ERP systems is here analysed in depth, informing the design considerations for modelling complex business systems, hosted within a cloud computing model.

The steady growth of the cloud services market within business is identified within the business drivers (Section 1.6). This provides an important backdrop that qualifies the importance of the research and the subject matter of complexity within these business services.

1.3 Problem Statement

Complexity within business and its supporting systems is closely interlinked with the evolution of cloud computing and Enterprise Resource Planning (ERP) systems. Business complexity is therefore a key factor in strategic technology investments. It is observed in practice however, that complexity is primarily a qualitative consideration within the early stages of current ERP implementations.

Within the innovation of ERP and the SaaS model, where the provider hosts the application and data, and service consumers access a front-end portal, complexity has a direct impact on a business transition to Cloud hosted SaaS solutions, as this has an impact on risk, cost and requirements. As a result, this SaaS model has been applied to flat business services to-date (Perrone, et al. 2010). Whilst more advanced business models are generating interest, work by Sarrell (2010) suggests that application of SaaS to ERP systems could pose several problems, not only technological, but also business and legal.

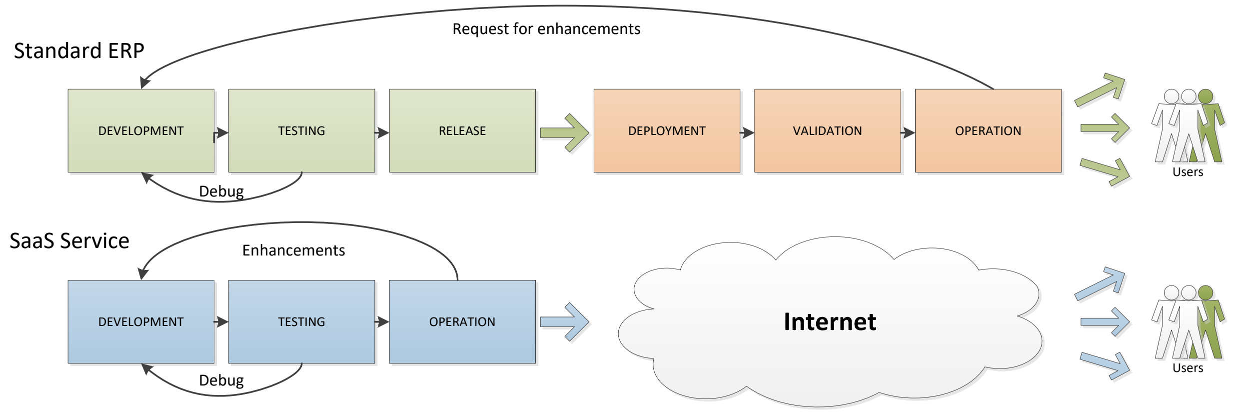


Figure 1: Standard Deployment vs. SaaS (adapted from Vass, 2012)

Business system procurement and solution implementation models are also changing. Figure 1 shows a representation of how traditional ERP system implementations differ from SaaS solution delivery. As solutions transition to a multi-tenant SaaS model, service provider evaluation shifts away from a dependency on that provider responding to a request for proposal (RFP) during the procurement process. Rather, the responsibility becomes with the service consumer, for gaining a deeper understanding of the SaaS solutions available, and evaluating which services and providers to subscribe to. Majority of information currently available to the service consumer is however a mesh of sales rhetoric and analyst based subjective reviews. This results in additional risk when a complex business is making the critically strategic decision of transitioning their internal systems to the Cloud.

1.4 Research Aims and Objectives

The key aim was to investigate ways to facilitate the transition of complex business and their underlying ERP systems to cloud hosted models of service. To mitigate the problems identified, the core **research question** for this thesis was synthesised as:

How to facilitate transition of complex business to SaaS cloud hosted services?

This research question was further expanded into the following focused **research aims,** directed at identifying:

- How has SaaS developed and how is it evolving? (Section 3.1)
- How has ERP developed and how is it evolving? (Section 4.2)
- What functional and technical approaches are being undertaken by key market players in this respect? (Section 3.2.1)
- What defines complexity in a business and how to quantify it? (Section 5.2)
- How to compare compliance of various ERP solutions to business requirements? (Section 8.2.3)
- How to evaluate validity of a range of SaaS service providers to a business? (Section 8.2.4)
- How can a technology service provider minimise risks and costs when developing a SaaS solution for a specific market? (Section 9.3)

This study therefore contributes to understanding cloud computing for commercial applications of the technology within a more complex business environment. The initial literature review served to understand the evolution of the technology, both architecturally and functionally. This enabled a forward view for ERP and the direction each of these characteristics was likely to evolve to. As a result, the requirement was identified for a toolset to support the strategic decision making for a business in making this technological transition. This led to the need for the development of a framework, through which the SaaS model could be evaluated, providing a new metric by which business complexity can be calculated and compared to available cloud solutions (refer to Section 5.1). This framework was additionally aimed for use to evaluate the feasibility of an industry by a service provider providing cloud managed solutions.

The key **objectives** for this work can be summarised as follows:

1. To investigate cloud applications and SaaS characteristics.
2. To investigate the evolution of the ERP infrastructure as an example of a complex business system.
3. To define the complexity in business systems, identifying an alternative approach for parameters distinguishing businesses as complex within this context.
4. To develop a framework and provide a scientific approach to the evaluation and decision-making process for the deployment of complex business systems over SaaS, by business or service providers alike.
5. To compile business case studies, using factual anonymised business datasets, extracted from companies characterised as having traditional complex enterprise support systems.
6. To apply the framework to these business cases studies, using these as proof of concept; analysing and evaluating results to derive their complexity and implementation risks.

1.5 Research Hypothesis

The author proposes to mitigate the subjective nature of a SaaS procurement process through modelling the structure of a business and quantifying its complexity. Additionally, by extending this complexity modelling to the compliance of SaaS business solutions on offer, this would enable a scientific evaluation and comparison of the service providers whilst minimising the associated risks and costs. Accordingly, in this thesis the author offers a practical solution to this process of decision making and analysis.

1.6 Market Drivers for Research

Worldwide spending on enterprise application software grew by 7.5%, reaching $149.9 billion in 2015, and is expected to increase to more than $201 billion in 2019, with accelerating cloud adoption driving new software sales (Forbes, 2015). Gartner's (2015) analysis of enterprise software investment shows that alternative consumption models to traditional on-premises licenses are accounting for more than 50% of new software implementations. These include SaaS, hosted licenses, on-premises subscriptions and open source.

As market statistics and forecasts described in Section 1.7 demonstrate, the cloud services industry is growing steadily. An understanding of the scale of this growth is relevant to highlight how technology customers and service providers need to formulate a cloud strategy, as cloud services become an inevitable reality for business in general.

Many business information technology (IT) strategies involve modernisation of core enterprise applications. A survey by Gartner (2015) found that 45% of respondents indicated that modernisation of installed on-premises core enterprise applications was one of the current top five IT project priorities. A further 41% indicated that "extending capabilities of core enterprise applications" was also a top priority. The key drivers for a service provider is therefore to bring to market a cloud portfolio to:

- Build platforms and solutions covering more complex business requirements.
- Secure a new sustainable revenue stream that adds material value to any legacy on-premises ERP offerings.

- Add marketable options on how ERP can be implemented and licenced to increase the breadth of customers these attract.
- Strengthen position in market for new market opportunities and to strengthen existing propositions.

This work references these statistics to rationalise the technology transition strategies identified for IT decision makers in Chapter 9, and how a framework could facilitate the decision-making process in support of a cloud business plan.

1.7 Market Opportunity for Research Application

Small and medium sized businesses or enterprises (SMBs or SMEs), may be defined as companies with 1 to 250 employees. SMB categories include micro 1 to 9 employees, small 10 to 49 employees, and medium 50 to 250 employees (Parallels, 2015).

Statistics by Parallels (2015), provide insight into the global cloud opportunity. The overall global SMB cloud market in 2015 was $98.7 billion. Figure 2 below shows how this is split between cloud services; Infrastructure-as-a-Service (IaaS), Web hosting, Unified Communications and Business Applications.

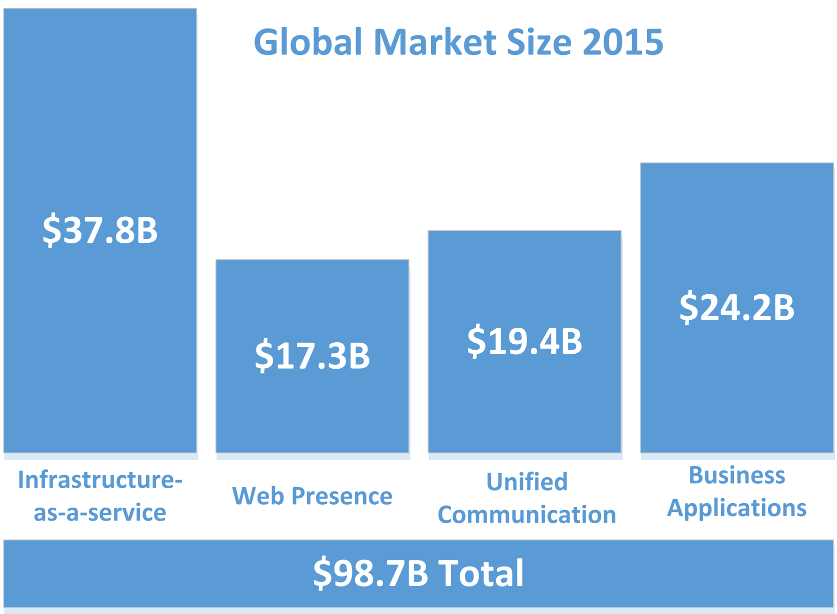


Figure 2: Global Market Size 2015 (adapted from Parallels, 2015)

Geographically, the 2015 Global SMB Cloud market was subdivided as follows:

Table 1: Global SMB Cloud Market (adapted from Parallels, 2015)

Region	Market Size
North America	$36.8 billion
East Asia, South Asia & Pacific	$23 billion
Middle East and North Africa	$3.1 billion
Europe and Central Asia	$26.2 billion
Latin America & Caribbean	$7.1 billion
Sub-Saharan Africa	$2.3 billion

The global market for IaaS is set to grow at a 14.1% compound annual growth rate (CAGR) to $56.2 billion in 2018. This is expected to be at a 16.7% CAGR in Europe and 8% CAGR in North America through to 2018.

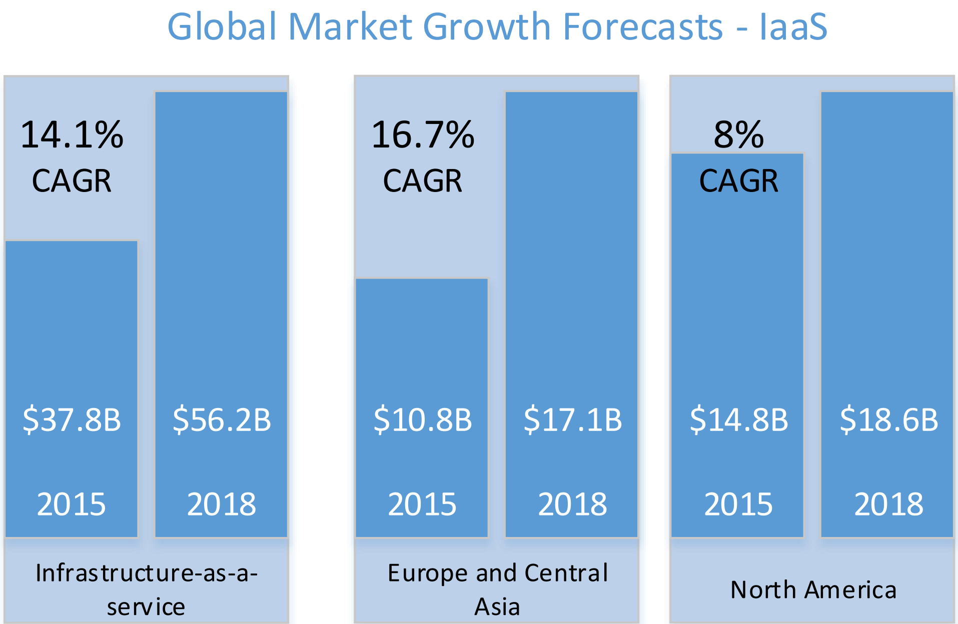


Figure 3: IaaS Forecasts 2018 (adapted from Parallels, 2015)

Similarly, the global market for Cloud Business Applications is set to grow at 25.9% CAGR to $48.3 billion in 2018. More specifically, the growth forecast is 21.9% CAGR in Europe and 25.9% CAGR in North America though to 2018, as highlighted in Figure 4.

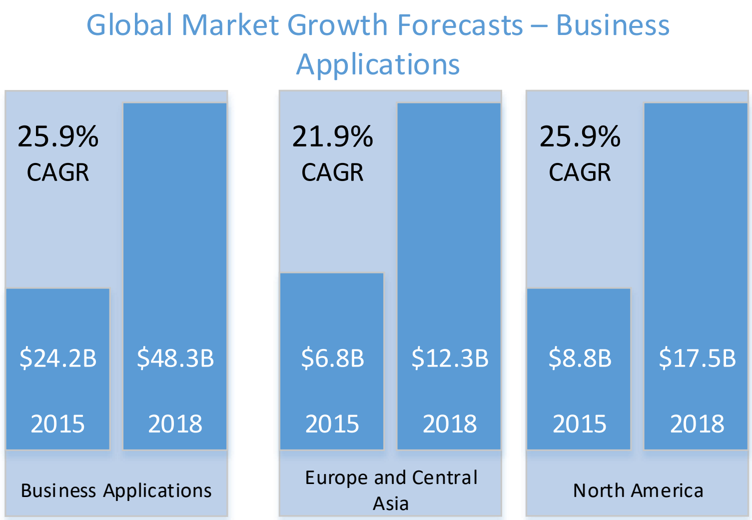


Figure 4: Cloud Business Applications Forecasts 2018 (adapted from Parallels, 2015)

In addition to these forecasts, Parallels (2015), report that in 2015 the average annual spend on IaaS per SMB was $9,582 in North America, and $5,936 in Europe. For Cloud Business Applications, the average annual spend per SMB was $4,872 in N America, and $3,548 in Europe.

2018 SMB Cloud Market Forecast

The overall global SMB cloud market opportunity is set to grow by 17.2% CAGR to $158.9 billion by 2018. Regionally, in North America this will be $51.8 billion (12.1% CAGR) by 2018, whilst within Europe this is set for 17.8% growth to $42.9 billion.

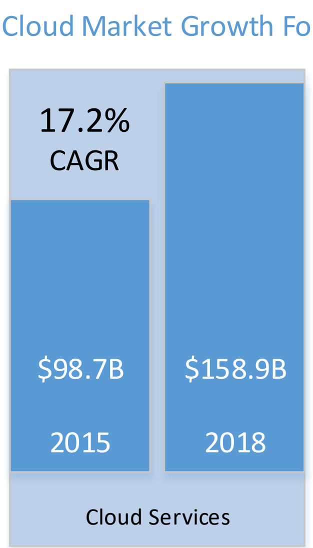


Figure 5: Cloud Market Forecasts 2018 (adapted from Parallels, 2015)

As the cloud market continues to evolve, market analysts Parallels (2015), recommend that service providers should focus on managed services and targeting businesses at the right time, with the right bundles, as the best ways to succeed in the next few years.

1.8 Thesis Structure

This section presents an overview of the thesis organisation. A short description of each chapter follows.

Chapter 1: Introduces the work undertaken with a short overview of the background, market drivers and the specification of the problems. The research aims, objectives and hypothesis are described, outlining the importance of this work within the context of the market drivers pushing the growth of cloud services. Here statistics from market analysts are detailed, highlighting the commercial interests that could benefit from this work.

Chapter 2: Explains the high-level research methodology and the various methods applied throughout this work. The chapter then proceeds to detail the approach taken for the background research and literature review, together with the tools, skills and techniques used.

Chapter 3: Drawing extensively from the background research, the evolution of cloud computing is explained in detail, progressing to in-depth analysis of cloud architecture and deployment models. This serves to formulate a forward vision as to the direction the computing industry is evolving to.

Chapter 4: Analysis the evolution of ERP as exemplars of complex business systems. The various ERP architectural changes occurring over the years are considered in parallel to the computing paradigms discussed in Chapter 3. This analysis provides the means to extrapolate on the architectural trends, to propose a solution for cloud based complex business systems.

Chapter 5: Following the analyses carried out in Chapters 3 and 4, the new SaaS Complexity Framework (SCF) is introduced, describing the various components making-up the framework. This chapter elaborates on the importance of defining and measuring business complexity, explaining the development of the SaaS Complexity Metric (SCM), and analysing the metric validation models required to validate the framework.

Chapter 6: Following up closely from Chapter 5, the development of the SaaS Complexity Framework Unified Model (SCFUM) is explained in detail, this being another key component of the SCF framework. This enables a business, comprised of Process Nodes and Functional Elements, to be mapped within cloud hosted services.

Chapter 7: Describes the application and validation of the SCF framework. The framework is evaluated against business case studies, selected as factual examples of complex business. The methodology for applying, validating and calibrating the SCFUM is explained, and the case mapping results are analysed and validated.

Chapter 8: Extending on the results from Chapter 7, methods and algorithms showing the calculation of the business complexity for each case study are evaluated and discussed. These provide a quantified measure of complexity as a unit of SaaS Complexity Index (SCI). These results are interpreted within a business context and examples of business intelligence (BI) output are detailed.

Chapter 9: Based on the findings from Chapter 8, cloud transition strategy options are proposed, defined from the perspective of the cloud solution service provider. These options are analysed and evaluated for cost, risk and possible return on investment (ROI). This enables a strategic comparison, which then surmises how a hybrid holistic strategy would minimise the possible risks associated with such business transition.

Chapter 10: This concludes the findings of the thesis, making recommendations for future work and providing examples for commercial applications of the SCF framework.

CHAPTER 2

RESEARCH TOOLS AND TECHNIQUES

2.1 Introduction

The methodology adopted for this research is outlined here, describing the methods applied for expanding knowledge on the research question and related objectives. This chapter describes the systematic approach taken in building upon the background research to develop a framework for modelling business within SaaS, explaining how the background research and literature reviews were undertaken. This methodology is expanded over the next chapters, to detail the specific steps undertaken in the execution of this work.

2.2 Methodology Overview

There are two key parts to the overall methodology. The **first part** outlines the methods related to the investigation of the research questions, background research execution and related literature review. The **second part** focuses on the methods applied for the development of the new framework, and its related validation. Figure 6 diagrammatically outlines the systematic methodology applied across the overall research.

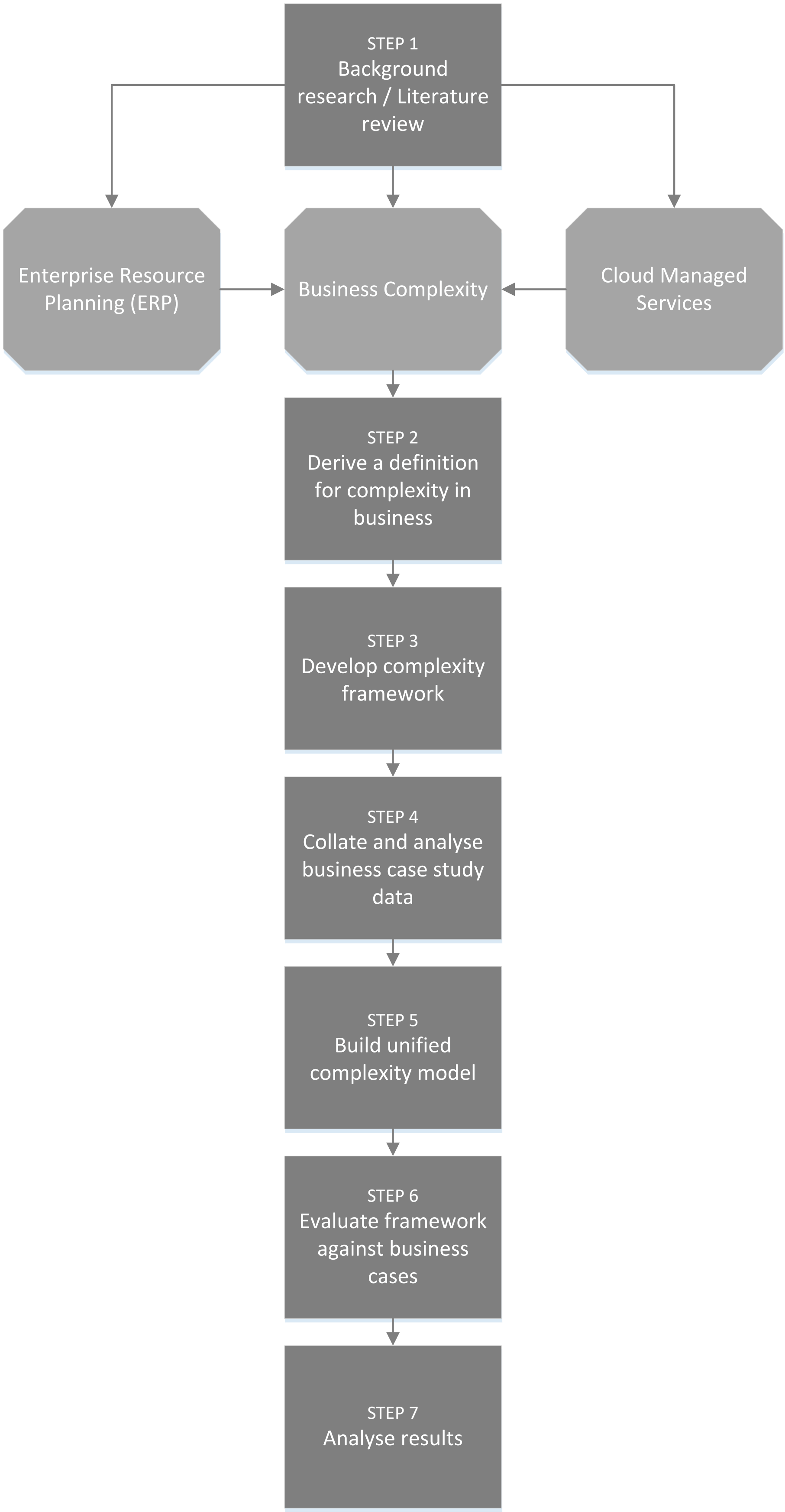


Figure 6: Research Methodology

The various steps within this methodology are structured as follows:

1) **Background research and literature review** - this step provides the background context of the study, and is an iterative process throughout the research, following the advances in technology being made by the industry. This also serves to extrapolate a forecast on where the ERP cloud architecture is evolving to, and in resolving some of the issues identified within the technology when applied to complex business systems. As described later in this chapter, the key context of this review is ERP systems, business cloud managed services, complexity in business, and business complexity models.

2) **Derive a definition for complexity in business** - the review of existing business complexity models found that these did not yield an applicable definition of business complexity, that could serve as a cross-over between business and ERP systems. Therefore, this identified the requirement to derive a new definition of business complexity, thus enabling the development of a new complexity framework in the next stages of the methodology.

3) **Develop a complexity framework** - this next step involves the development of a new framework to support the transition of a complex business system to a cloud managed service model. The development of this framework has several facets, including the derivation of a metric to measure business complexity, based on the definition of complexity described in the previous step of the methodology.

4) **Build a unified complexity model** - another element in the development of the framework is the dissection of a sample of traditional ERP systems. Building upon the concepts defined within the previous steps of methodology, this leads to the development of a unified complexity model. This unified model is the means through which the complexity metric can be calculated and quantified.

5) **Collate and analyse business case study data** - to validate the new framework, anonymised data from several businesses is collected and analysed. Each business selected is to be based on the definition of complexity specified in the previous steps. The collated datasets serve as exemplars of factual complex business, and form the case studies against which the framework can be validated and calibrated in latter steps.

6) **Evaluate framework against business cases** - the complexity framework, and the related complexity unified model, are then applied against each dataset for all the business cases. The objective of this step is to evaluate the effectiveness of the framework as a tool within a commercial application. The results are analysed in detail, enabling the framework to be fine-tuned based on the initial findings.

7) **Analyse case results -** the resultant data, generated through the application of the complexity framework would provide a quantified measure of complexity for each business. This allows the business cases to be analysed within a cloud managed services context. Several calculations for comparative analyses are made possible, such as median complexity and baseline values, amongst others.

Later chapters describe in more detail the specific methods applied within each step, and expand upon this research methodology.

2.3 Literature Review

A literature review was undertaken as a preliminary investigation to inform this methodology and providing the theoretical framework for this thesis. Using the Wallace and Wray (2006) categories of literature, two key types of literature were found to be most applicable, these being:

1) Practice literature – covering business research and development white papers, and publications from the industry’s own research outcomes and developments.
2) Research literature – covering books, academic publications, peer research findings and conference proceedings, amongst others.

The wider research methodology itself took the form of a quantitative, empirical study, with some qualitative elements. The business cloud services market over the duration of the research was evolving at a pace, and therefore the literature reviews were ongoing throughout the research study.

The methodology applied in the literature review was that of a systematic search for a principally thematic review, with chronological elements. The importance of the time element was due to the fast pace of the developments in this computing sector, which placed an additional focus on chronology within the evolution of the research, and in shaping technological forecasts within an iterative approach. To better align the literature search requirement within the structured search, the following keyword hierarchy was applied:

- SaaS + ERP + Cloud Computing
- Enterprise Resource Planning (ERP)
- Business Systems
- Complexity in Business
- Complex Business
- Complex Business Systems
- Measuring Complexity
- Complexity

The search methods comprised a range of sources, including: electronic searches, Google Scholar, electronic databases, brick and mortar library, and industry white papers. The outcome of each search was then reviewed, and weighed for validity and for justification in use. Each search iteration output across the duration of this work was then combined within the wider research findings.

2.4 Methodology Skill Sets and Toolsets

Various skills and research tools have been used within the methodology. Key of these were the following:

Professional Background

The author's professional background as a Management and Information Technology (IT) consultant, with years of practical business systems implementation experience, served underpin the whole research. Having driven system deployments and business process change within multi-national organizations, this expertise enabled direct recognition of the impact business complexity has in these scenarios, as well as providing access to the business case data. Moreover, as a solution architect for projects across 13 business sectors and 32 different industries, this provided a knowledgebase to inform methods that were industry-agnostic.

Microsoft Office Professional 2013

The Office suite of applications served several purposes:

- **Excel** for the manipulation of the business case data, and the generation of pivot tables for summarisation of results.
- **Word** for representation of results and outlining the research thesis.
- **Visio** for the graphical representation of concepts and diagrammatic illustration of ideas.

Microsoft Dynamics ERP

Microsoft Dynamics AX (2012 R3) and NAV (2009 R2) systems were used as general exemplars of ERP systems within the research methods, and served for comparative analysis in the framework validation process. This additionally required access to their related SQL database back-end. The rationale for selection of the ERP systems was based on their widespread use within the global business community, as well as due to ease of access through the author's professional background. Further details on these systems are provided in Sections 4.4 and 4.4.2.

VMWare Virtualization

VMWare Workstation (v10.0.7) was used to set up and manage several virtual machines (VMs) on a single hardware platform, running the Windows 7 Ultimate SP1 operating system (OS). The VMs were generated to host a combination of Linux Ubuntu (15.10), Windows 7 and Windows 10 OSs. This virtualization software suite served as a test environment for the application of concepts encountered during the background research.

Hardware

The following computer hardware was used throughout this study:

- An HP EliteBook 8560w laptop running Windows 8.1 Pro on a 64bit system, with an Intel i5-2540 CPU and 8GB RAM. This environment was used for data manipulation, summarisation and reporting, as well as the findings write-up. Moreover, it was used to access various VM environments for the source business cases data, as well as the VMWare test environments.
- A Lenovo ThinkPad laptop running Windows 7 Ultimate SP1 on a 32bit system, with an Intel i5-2520 CPU and 4GB RAM. This environment hosted the VMWare virtualization suite, and ran the vCenter server, broadcasting the VM test environments.

Microsoft Cloud

The Microsoft OneDrive Cloud file-hosting service, was used to host the business case datasets, as well as any other elements related to this research. Data manipulation was carried out through direct synchronisation with this service, ensuring automatic redundancy of the datasets and documents.

2.5 Summary

The research methodology in this chapter outlined seven key steps making-up the structure of this research. The skill sets and toolsets applied throughout this work were then explained. Describing the methodology in more detail, the methods applied for each relevant step are detailed in subsequent chapters as follows: the derivation of a definition for complexity in business, and the development of a complexity framework, in Chapter 5; the building of a unified complexity model in Chapter 6; the application of the business case study data, and the evaluation of the framework in Chapter 7; and finally the analysis of the case results in Chapter 8.

CHAPTER 3

BACKGROUND RESEARCH AND LITERATURE REVIEW

3.1 Evolution of Cloud Computing

Computing has six generally accepted computing paradigms, covering a span of time from the initial days of mainframe computing to personal computing, network computing, Internet computing, grid computing and finally cloud computing. These six paradigms are represented below in figures adapted from Voas and Zhang (2009). As computing systems are in constant evolution, this chapter investigates how computing paradigms have changed over time, demonstrating how cloud concepts have existed for many years, evolving gradually over time.

This being Step 1 in the methodology, the objective here was that of extrapolating on these computing architectures to propose the next future model in the evolution of the computing industry, required to support more complex SaaS. From Section 3.2.1 onwards, cloud computing services, its architecture, and deployment models, are examined in detail, allowing for these concepts to be applied in later chapters.

3.1.1 Mainframe Computing

The term "mainframe computer" is used to distinguish very large computers used by institutions to serve multiple users from the personal computers used by individuals. Mainframe computers are capable of handling and processing very large amounts of data very quickly, much more data than a typical individual requires to work on their own computer. (Zandbergen, 2015)

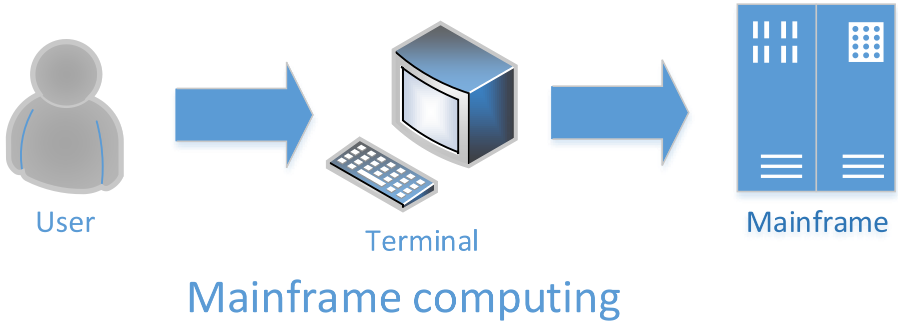


Figure 7: Mainframe Computing (adapted from Voas and Zhang, 2009)

As described by Ceruzzi, P.E. (2003), the mainframe computers which were developed in the 1950s, have continued to evolve and are still in use. One of the characteristics of early mainframes was that they did not have the typical interactive interface as is used with personal computers. Early mainframes accepted various modes of input, and later versions typically had a dedicated terminal, not unlike a modern display monitor with a keyboard.

The IBM Archives (2015) detail how a more interactive form of computer use developed commercially by the middle 1960s. In a time-sharing system, multiple computer terminals let many people share the use of one mainframe computer processor. This was common in business applications and in science and engineering. End users generally did not directly interact with the machine, but instead would prepare tasks for the computer on off-line equipment, such as card punches. Several assignments for the computer would be gathered up and processed in batch mode. After the job had completed, users could collect the results. In some cases, it could take hours or days between submitting a job to the computing centre and receiving the output. These limitations led computer scientists to break down complicated tasks into manageable ones so that various parts of a problem could be worked on in parallel.

The name “terminal” arose in the 1960’s to describe devices used to access time-shared computers. Originally these were no more than a keyboard and printer, and worked like a teletype machine. These implied users would type a command and the computer typed a response. This was far more interactive than punching cards, running them through the computer, then getting a printout.

The IBM Archives (2015) further detail how the usefulness of computers was greatly expanded by the introduction of IBM data transmission terminals, enabling distant locations to communicate with a central computer to enter or retrieve information. This ability meant that information stored in the system could be automatically updated as transactions occurred, and made available upon request to headquarters management as well as field personnel.

Over the years, advances in mainframe performance have been made possible through revolutionary developments in microelectronics, innovations in processor architecture and in operating systems architecture, all of which having played a significant role.

Mainframe systems can be used by large number of users. This implies that in a large organisation, individual employees can sit at their desk using a personal computer, but they can send requests to the mainframe computer for processing large amounts of data. A typical mainframe system can support hundreds of users at the same time. The actual hardware components inside a mainframe computer are similar in type to what personal computers would use: motherboard, central processing unit and memory. For a mainframe however, these components would be a lot more powerful and expensive than those found in personal computers. (IBM Archives, 2015)

3.1.2 Personal Computing

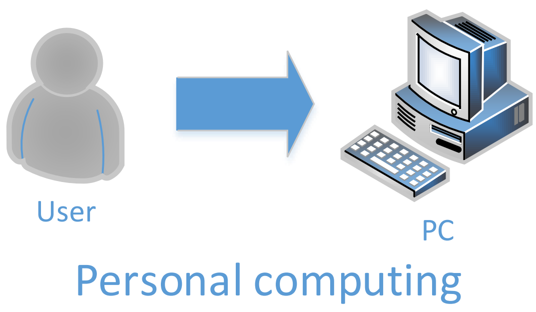


Figure 8: Personal Computing (adapted from Voas and Zhang, 2009)

Reimer (2005), Shapiro (2000) and others, describe how the history of the personal computer as mass-market consumer electronic device effectively began in 1977 with the introduction of microcomputers, although some mainframe and minicomputers had been applied as single-user systems much earlier. A personal computer was one intended for interactive individual use, as opposed to a mainframe computer where the end user's requests are filtered through operating staff, or a time-sharing system in which one large processor is shared by many individuals. After the development of the microprocessor, individual personal computers became affordable consumer goods. This formalised the definition of a personal computer (PC) as a general-purpose computer, whose size, capabilities, and original sale price made it useful for individuals, and which is intended to be operated directly by an end-user with no intervening computer operator.

The base of the PC was a hardware layer consisting of a processor, connected to a variety of input, output, and memory devices, such as disks and screens. The operating system resided on top of this hardware layer, allowing applications to interact with the lower level functions of the hardware layer. Applications, such as word processors and spreadsheets were stacked on top of the operating system, as were local data and files. The entire environment of the desktop was therefore contained and could be operated independently without any network connection to another computer (IBM Archive, 2015).

As desktop computing rapidly grew increasingly complex through more sophisticated programs, operating systems, and networking, the cost of corporate computer system support grew exponentially. These issues, multiplied by thousands of users, have driven corporate IT leaders toward radically diverse solutions for the desktop computer to control costs, improve the quality of the computer experience for the end user, and increase efficiency within the workforce (Raimer, 2005).

3.1.3 Network Computing

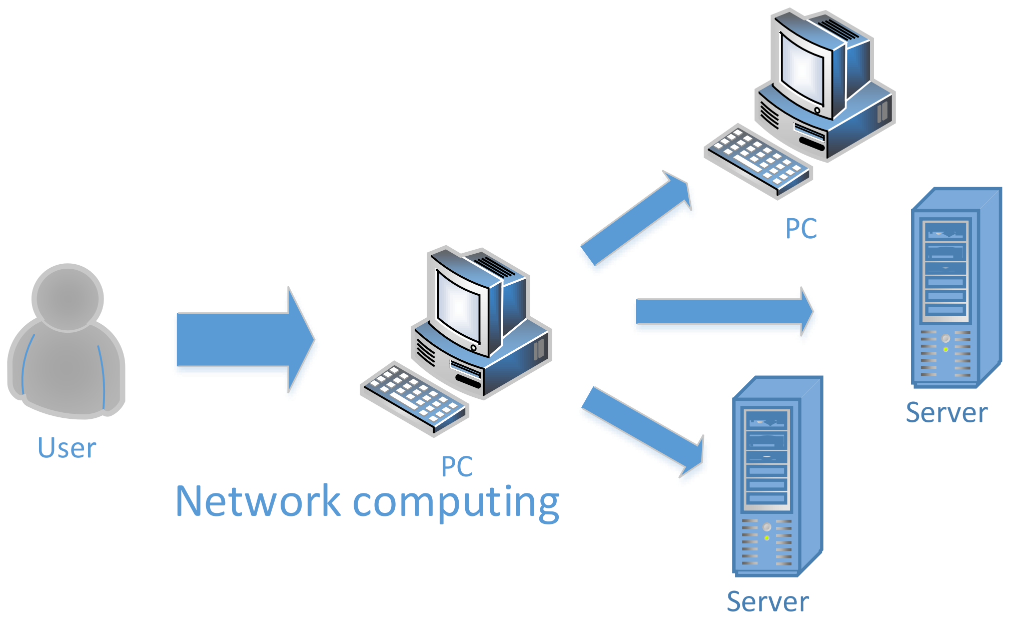


Figure 9: Network Computing (adapted from Voas and Zhang, 2009)

Network computing refers to “the use of computers and other devices in a linked network, rather than as unconnected, stand-alone devices” (Technopedia, 2016). As computing technology has progressed during the last few decades, network computing has become more frequent, especially with the creation of cost-effective and relatively simple consumer products such as wireless routers, which turn the typical home computer setup into a local area network (Technopedia, 2016).

A computer network is therefore a collection of computer equipment, connected with wires, optical fibres, or wireless links so the various devices (known as nodes) can communicate to one another and exchange data (computerised information).

In network computing, computers often share broadband and other resources. Many larger business networks also share hard drive space, where any networked computer has access to the same data through a server or another hardware setup. Networking became a more efficient way to deliver more functionality to large number of computers or devices. For example, a network could allow for lower software licensing fees than buying the software for a specific number of stand-alone devices.

During the 1980s and 1990s, everything became much more standardized and it is now possible to connect virtually any machine to any other and get them exchanging data without too much effort. This is largely because most networks now use the same system, called "Ethernet" (Woodford, 2016).

The frame delivering the message goes to every machine and device on the network. Each machine reads the destination address to figure out whether the frame is intended for them. If so, they act on it; if not, they ignore it. Any machine on the network can transmit messages through the ether at any time.

Client-Server Networking

The client-server model was developed to allow more users to share access to database applications. Compared to the mainframe approach, client-server offered improved scalability because connections could be made as needed rather than being fixed. The client-server model also supports modular applications that can make the job of creating software easier. Mobile devices as well as desktop computers can both function as clients.

In the two main types of client-server systems, "two-tier" and "three-tier", software applications are separated into modular pieces, with each piece installed on clients or servers specialised for that subsystem. A server device typically stores files and databases including more complex applications.

Although the client-server model could be used by programs within a single computer, it is more significantly an important concept for networking. In this case, the client establishes a connection to the server over a network, which could be either a LAN or the Internet. Once the server has fulfilled the clients' request, the connection is terminated. Computer transactions in which the server fulfils a request made by a client are very common, and the client-server model is certainly one of the central ideas of network computing (Tassabehji, 2003, pg.43).

The client-server model organizes network traffic per client application as well as per device. Network clients make requests to a server by sending messages, and servers respond to their clients by acting on each request and returning results. One server generally supports numerous clients, and multiple servers can be networked together in a pool to handle the increased processing load as the number of clients grows.

A drawback of client-server systems is that the server is a single point of failure (Singh, 2004, pg. 803). It is the only component with the ability to dispense the service. There can be any number of clients, which are interchangeable and can come and go as necessary. If the server goes down, however, the system stops working. Thus, the functional abstraction created by the client-server architecture also makes it vulnerable to failure.

Alternative Program Relationship Models

Other program relationship models included peer-to-peer (P2P) and master-slave. In the P2P model, each node in the network can function as both a server and a client. This treats all devices as having equivalent capability rather than specialized client or server roles. Mitchell (2016) notes that compared to client-server, peer to peer networks offer some advantages such as better flexibility in expanding the network to handle large number of clients. Client-server networks generally offer some advantages over peer-to-peer as well, such as in the ability to keep data protected from attackers.

In the master-slave model, one device or process (known as the master) controls one or more other devices or processes (known as slaves). Once the master-slave relationship is established, the direction of control is always one way, from the master to the slave (Rouse, 2008).

These advances have led to the more recent developments which have made network computing more sophisticated. Processes such as network virtualization, where hardware networks may be logically partitioned, as well as cloud computing, where the shared network resources can be located remotely for greater data security.

3.1.4 Internet Computing

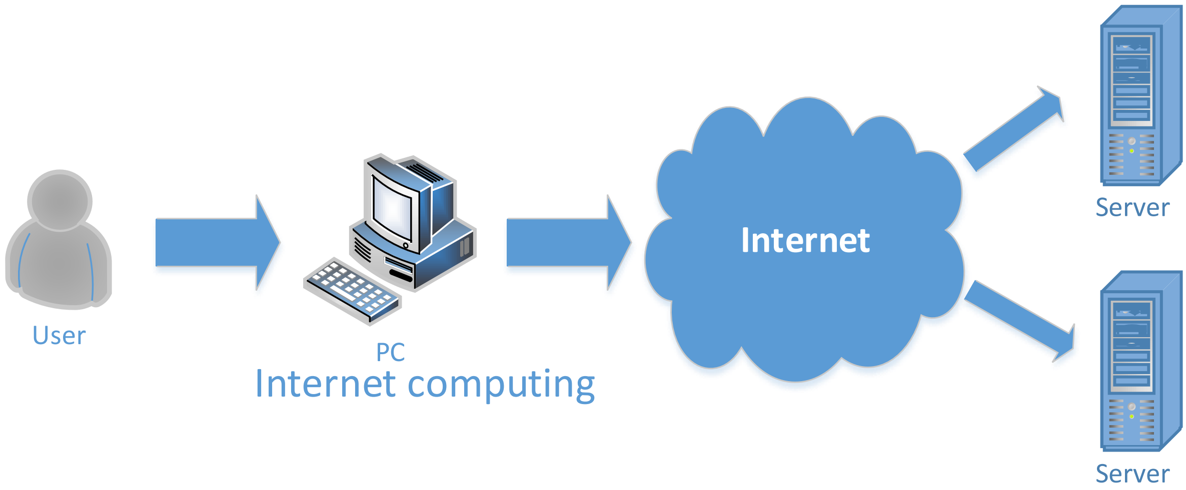


Figure 10: Internet Computing (adapted from Voas and Zhang, 2009)

The concept of a global network of interconnected computers had surfaced as early as the 1960s. This was eventually made possible through the technological evolution that began with early research on packet switching and the ARPANET related technologies. As detailed by Kahn and Cer (2003), the term “Internet” refers to the global information system that:

(i) Is logically linked together by a globally unique address space based on the Internet Protocol (IP) or its subsequent extensions/follow-ons.

(ii) Can support communications using the Transmission Control Protocol/Internet Protocol (TCP/IP) suite or its subsequent extensions/follow-ons, and/or other IP-compatible protocols.

(iii) Provides, uses or makes accessible, either publicly or privately, high-level services layered on the communications and related infrastructure.

The internet is in effect the network of networks, that consists of millions of private, public, academic, business, and government networks of local to global scope and linked by a broad array of electronic, wireless, and optical networking technologies. The internet carries an extensive range of information resources and services, such as the interlinked hypertext documents, applications of the World Wide Web (WWW), electronic mail, telephony, and peer-to-peer networks for file sharing. This is a global network comprising many voluntarily interconnected autonomous networks, operating without a central governing body (Leiner, 2012).

The internet continues to grow, driven by ever greater amounts of online information, knowledge, commerce, entertainment and social networking. As at 2016, internet use covered 50.1% of the global population (Internet World Stats, 2016). This growth is often attributed to the lack of central administration, which allows the organic growth of the network, as well as the non-proprietary nature of the internet protocols, which encourages vendor interoperability and prevents any one company from exerting too much control over the network (Coffman and Odlyzko, 1998). This structure was found to be highly robust to random failures, yet very vulnerable to intentional attacks.

3.1.5 Grid Computing

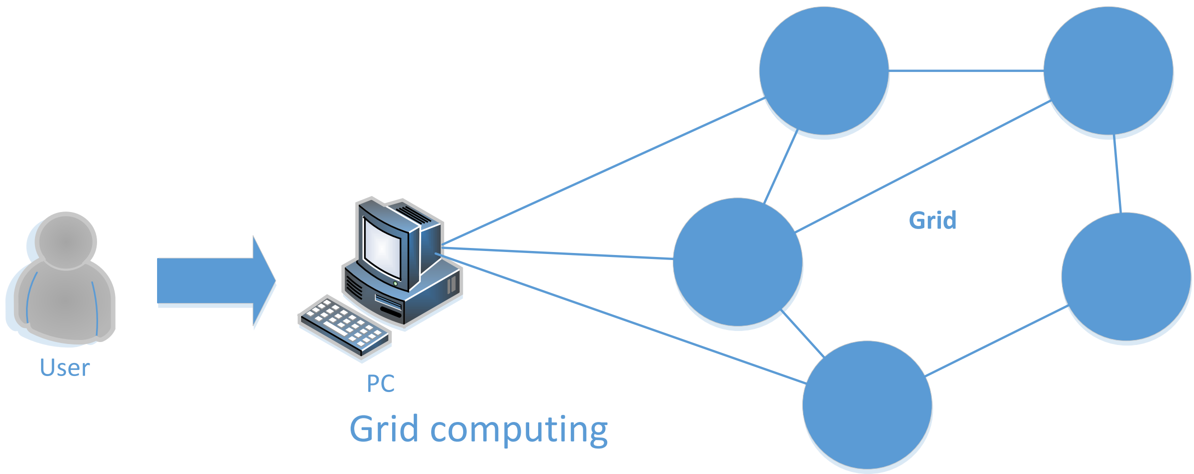


Figure 11: Grid Computing (adapted from Voas and Zhang, 2009)

Grid computing is a distributed architecture where a large number of computers are connected to solve a complex problem. In the grid computing model, servers or personal computers run independent tasks and are loosely linked by the Internet or low-speed networks. Computers may connect directly or via scheduling systems (Rouse and O'Reilly, 2015).

This computer networking model allows data, programs, and storage devices of the connected computers to be shared, as well as pooling their data processing power. In this architecture, an extremely complex or large problem is portioned out to potentially thousands of computers connected through the internet. Each computer's processing power is used only when it is on but idle, such as when its user is reading the screen or typing on the keyboard. This technique results in huge cost savings over buying supercomputers, and is generally more reliable because it is unaffected by the failure of even a large percentage of connected computers. This model requires sophisticated software and powerful servers to systematically farm out the workload and collect and assemble the results (BusinessDictionary, 2015).

Dabrowski (2009) finds that an application of grid computing therefore needs to ensure high redundancy, so a robust failure recovery is built into the model, since it is highly probable that some number of compute nodes will disconnect or fail. Security is another issue with this architecture, as the controls on member nodes are usually very loose. As a result, the grid computing model works well for only a narrow subset of applications.

Patni (2011) describes how one of the main strategies of grid computing is to use middleware to divide and apportion pieces of a program among several computers, sometimes up to many thousands. Grid computing involves computation in a distributed fashion, which may also involve the aggregation of large-scale clusters. The size of a grid may vary from small, confined networks within a corporation, to large, public collaborations across many companies and networks. The notion of a confined grid may also be known as an intra-nodes cooperation; whilst the notion of a larger, wider grid may thus refer to an inter-nodes cooperation.

Distributed Computing

Distributed computing is characterised as a subset of the grid computing model. In this model, components of a software system are shared among multiple computers simultaneously to improve efficiency and performance. The computers interact to achieve a common goal, by sending each other messages.

In the enterprise, distributed computing has often meant putting various steps in business processes at the most efficient places in a network of computers. For example, in the typical distribution using the 3-tier model, user interface processing is performed by the PC at the users' premises, business processing is done on a remote computer, and database access and processing is conducted in another computer that provides centralized access for many business processes. Each of the three computers could in effect be running a different operating system (Rouse, 2015).

For software to make best use of this architecture, the actual calculations need to be broken up into separate elements that can be run on several computers. Some calculations may be entirely sequential, with each step following the completion of the other. Other calculations may however be able to execute in parallel, allowing these to be broken up into elements and running them separately but at the same time, subsequently combining the results. This adds complexity to software development, as development is not limited to the calculations to solve the task at hand, but also the ability to break it up into elements that can be run separately. A special controller software is therefore needed to manage the resources of the various computers.

3.2 Cloud Computing

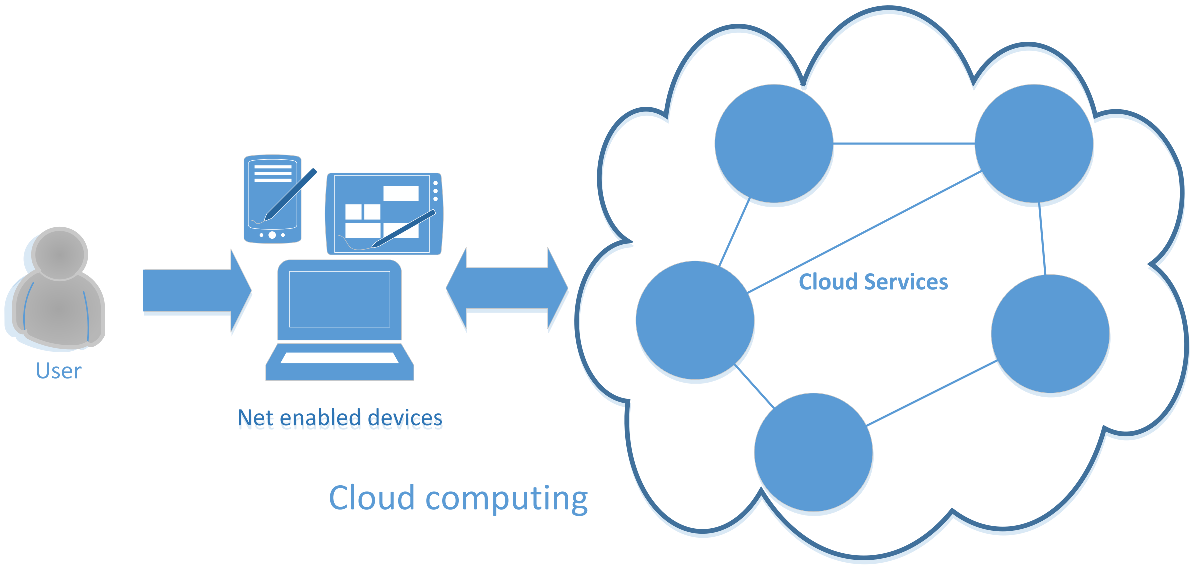


Figure 12: Cloud Computing (adapted from Voas and Zhang, 2009)

Cloud computing extends on the concepts developed through the other five paradigms, providing a model for sharing computing resources that include applications, computing power, storage, networking, development, and deployment platforms as well as business processes, based on an underlying internet foundation. As described by Buyya et al. (2011), the term "Cloud Computing" was devised as an abstraction of the internet and the technological infrastructure that supports it. The cloud computing model defines a method of computing which is dynamically scalable, where often virtualized resources are provided as a service over the internet.

The aim of the cloud model is to allow users to benefit from all these technologies, without the need for expertise within any one of them (Hamdaqa, 2012). Buyya, et al. (2011) further describe how this technology provides advantages such as cost savings, high availability, and easy scalability, whilst helping users focus on their core business instead of being impeded by IT. Research by Shawish and Salama (2014) shows this model therefore provides a big paradigm change in computing and has become a significant technology trend.

Kushida, et al. (2011) noted that there is an expectation that cloud computing will reshape IT processes and the IT marketplace. Cloud computing technology also allows users to consume these services through a variety of devices, ranging from PCs to smartphones, providing access to programs, storage, and application development platforms over the internet, via services offered by cloud service providers.

3.2.1 Cloud Services and Architecture

Cloud computing therefore exhibits parallel concepts to each of the other five paradigms. Examples of these cloud computing parallels across the paradigms are as follows:

- **Mainframe Computing**: The possibility for the use of connected dumb terminals for cloud service access.
- **Personal Computing**: End-users could still directly operate cloud services with no intervening computer operator, and these could interact with applications within the PCs own environment.
- **Network Computing**: Business and home networks still actively connect devices in local networks for running on-premises systems.
- **Internet Computing**: Cloud services could operate across networked systems, where this could be a hybrid of internet based and on-premises networks.
- **Grid Computing (Distributed)**: Cloud hosting services also enable sharing computing resources that include applications, computing power, storage,

networking, development, and deployment platforms as well as business processes, based on an underlying internet foundation.

The catalyst which brings together all the six computing paradigms is virtualization. Neto (2014) describes how virtualization has been driving the technology into this communication and information evolution. Using virtualization, it became possible to execute one or more operating systems simultaneously in an isolated environment. Complete virtual computers could be executed inside one physical hardware, which in turn can run a completely different operating system.

Cloud computing can therefore be viewed as a collection of services, which can be presented as a layered computing architecture, shown in Figure 13.

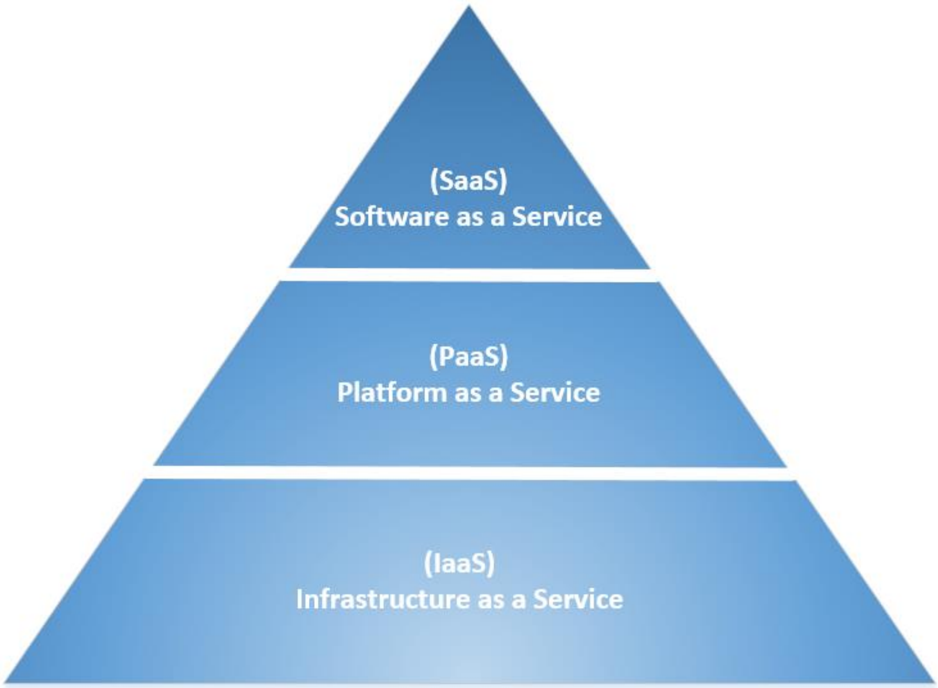


Figure 13: Cloud Services (adapted from Buyya, et al., 2011)

3.2.1.1 *Infrastructure-as-a-Service (IaaS)*

Infrastructure-as-a-Service (IaaS) is the leasing of infrastructure in the form of computing resources and storage, as a service. This implies virtualized computers with guaranteed processing power together with reserved bandwidth for storage and Internet access.

Buyya, et al. (2011) describe how the service consumer therefore gains access to a provision of processing, storage, networks, and other fundamental computing resources, with the ability to deploy and run whatever preferred software. This could include both operating systems and applications. The service consumer in this case does not manage or control the underlying cloud infrastructure, but has control over the operating systems, storage, and deployed applications together with the possibility of a level of control on some networking elements, such as host firewalls.

This service layer differs from Platform-as-a-Service (PaaS), as the virtual hardware is provided without a software stack. The consumer would in effect provide their own Virtual Machine (VM) image applied to one or more virtualized servers. IaaS is the cloud service that provides the closest outside access to a host physical infrastructure. Some examples of commercial IaaS providers are:

> **Amazon Elastic Compute Cloud (EC2),** released in 2006, is a central part of Amazon Web Services (AWS), which provides two key IaaS services, allowing consumers to either specify a specific VM operating system and application set, deploying their own applications on it; or for consumers to provide their own VM image to be execute on the servers. Payments are then based on compute time, storage, and network bandwidth (Amazon Web Services, 2016).

> **Microsoft Azure,** released in 2010, is Microsoft's cloud computing infrastructure, and platform for managing applications and services on Microsoft's hosting datacentres. Azure uses a specialised operating system to manage the cluster of hosting datacentres, referred to as the Fabric Controller. The fabric controller also controls reliability in the case of server crashes, and provides application memory resource and load balancing. Developers can interact with services via application programming interfaces (APIs) or through a web based Azure portal. This allows users to browse active resources, modify settings, launch new resources, and view basic monitoring data from active virtual machines and services (Microsoft, 2016).

> **Google Compute Engine (GCE),** released in 2013, is the IaaS component of the Google Cloud platform. The compute engine allows users to deploy their own VMs or launch standard images on demand. GCE can be managed through an API,

command line interface (CLI) or web console. The API provides administrators with a VM, domain name server (DNS) and load balancing capabilities. VMs are available in several CPU and RAM configurations. For security, data is encrypted using the AEC-128-CBC algorithm. GCE also allows administrators to select the region where certain data resources will be stored and used. This allows for the creation of advanced networks at the regional level, and supports some of the legal requirements of some countries for business data storage (Google, 2016).

Rackspace Open Cloud, released in 2008, is a scalable, high-performing cloud environment providing a combination of compute, network, storage, and traffic management services. Rackspace makes use of OpenStack, an open-source software platform consisting of interrelated components controlling hardware resource pools of processing, storage, and networking resources throughout a datacentre. The fabric controller for this service is called OpenStack Compute (Nova). OpenStack's architecture is designed to scale horizontally, with no proprietary hardware or software requirements, providing the ability to integrate with legacy systems and third-party technologies (Rackspace, 2016).

IBM SmartCloud Enterprise, released in 2007, is IBM's cloud services offering. At the foundation level, this consists of the infrastructure, hardware, provisioning, management, integration and security that serve as the basis for a consumer private or hybrid cloud. IBM offers cloud delivery options including solely private cloud, solely public cloud, and variations in between. Private, public and hybrid clouds are not strictly distinct, as IBM allows the option to build a customized cloud solution out of a combination of public cloud and private cloud elements (IBM Cloud, 2016). IBM offers five cloud provision models:

(i) Private cloud, owned and operated by the customer.
(ii) Private cloud, owned by the customer, but operated by IBM (or another provider).
(iii) Private cloud, owned and operated by IBM (or another provider).
(iv) Virtual private cloud services (based on multi-tenanted support for individual enterprises).
(v) Public cloud services (based on the provision of functions to individuals).

3.2.1.2 *Platform-as-a-Service (PaaS)*

Platform-as-a-Service (PaaS) is similar to IaaS but includes operating systems and additional services with a focus on a particular application. CSCC (2015), describe how the service consumer is provided with the capability to deploy onto the cloud infrastructure both their own or third-party applications created using programming languages, libraries, services, and tools supported by the service provider. The service consumer does not manage or control the underlying cloud infrastructure whether it is network, servers, operating systems, or storage. The consumer does however have control over the deployed applications and possibly the configuration settings for the application-hosting environment.

In addition to virtualized servers and storage, PaaS therefore provides operating systems and application sets, typically in the form of a virtual machine. This then provides access to required services such as a SQL databases and specialized local resources. PaaS is therefore IaaS with a custom software stack for the given application. The PaaS platform can be an entire virtualized platform that could include one or more servers, virtualized over a set of physical servers, operating systems, and specific applications such as Apache and MySQL for web-based applications. Whilst these platforms could be predefined by the service provider, businesses could provide their own VM image containing all necessary user-specific applications.

An example of a PaaS is the Google App Engine. The app engine is a service that allows users to deploy their web applications on Google's scalable architecture. The app engine also provides APIs for persistently storing and managing data, in addition to support for authenticating users, e-mail management and other server services. The environment in which the web application is executed however restricts access to the underlying operating system.

3.2.1.3 *Software-as-a-Service (SaaS)*

Software-as-a-Service (SaaS) is described by Buyya, et al. (2009) as the service model where the service provider provides the service consumer with the use of the provider's applications running on a cloud infrastructure. The applications are accessible from a numerous range of client devices, through either a thin client interface, such as a web browser or an actual program interface. The consumer does not manage or control the underlying cloud infrastructure, and has no access to the underlying network, servers, operating systems, storage, or even individual application capabilities, other than possibly some limited customer-specific application configuration settings. SaaS therefore allows consumers to lease an application and pay only for the time used.

Carraro and Chong (2006), describe how an early approach to SaaS was the Application Service Provider (ASP) model. ASPs provided subscriptions to software that was hosted or delivered over the internet, traditionally through the hypertext transfer protocol (HTTP). The service charge fees were then based on application use. Whilst SaaS is a similar concept, ASP did not have many of the benefits of multi-tenancy, such as cost sharing and economies of scale, to name a few. In effect, ASP would be more comparable to third-party hosting, rather than to true multi-tenant SaaS.

3.2.1.4 *Web Services*

Another approach to SaaS is the use of applications over the Internet that execute remotely. Such software could be in the form of services used by a local application, defined as web services, or a remote application accessible through a web browser. Typically, web services demonstrate the following features:

- These are typically small units of code.
- Designed to handle a limited set of tasks.
- Use XML based communicating protocols.
- Are independent of operating systems.
- Are independent of programming languages.
- Provide possibility to connect people, systems and devices.

An example of a remote application service is Google Apps, which provides several business applications through a web browser. This approach of remotely executing applications commonly relies on an application server to expose the needed services. An application server is a software framework that exposes APIs for software services, such as transaction management or database access. Examples include Red Hat JBoss application server, Apache Geronimo, and IBM WebSphere application server.

3.2.1.5 *Mixed Service Models*

Businesses rely on both purchased software and in-house developed applications. These applications could have services exposed to one-another, with the overall solution forming part of the on-premises platform. Many SaaS applications also frequently expose services that can be accessed by on-premises applications or by other cloud applications. This allows service consumers the possibility of creating their own on-premises applications, integrated with services exposed by SaaS applications part of the cloud platform. These mixed service models are also referred to as hybrid.

3.2.2 Layers of Cloud Computing

Cloud computing can be viewed as a collection of services, presented as a layered cloud computing architecture. The standard layering of these services, as offered by cloud service providers, are generally IaaS, PaaS, SaaS. There are common views as to how the architecture is structured, generally subdivided as "physical" and "abstraction" layers. A key element is that in such a layered architecture, each layer talks only to the layer immediately below it, and not to any other layer. This introduces an important reliance across layers, in that each layer must provide the stability and robustness to ensure reliability of service throughout the stack. The following is a general view of the cloud architecture layers, and how they relate to one another.

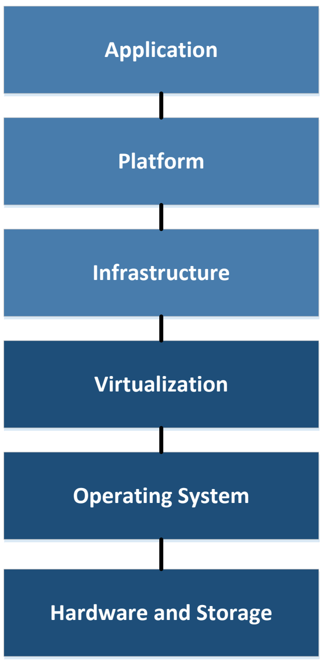


Figure 14: Architecture of Cloud Computing (adapted from Jones, 2009)

Application

This is the layer end users interact with. This contains the end-user applications such as SaaS, Business-Process-as-a-Service (BPaaS), and cloud service provider applications for cloud usage accounting and billing.

Platform

The platform layer provides for resource allocation covering load balancing and resource elasticity. Automated cloud management through API and command-line user interface (UI), are also available together with web UIs and other graphical user interfaces (GUIs) for cloud management. This layer also references access control for role-based cloud access with segmentation for users, tenants and administrators.

Infrastructure

This is the cloud foundation layer handling the management of:

- Virtual instances enabling starting, stopping and rebooting of VM instances.
- Virtual hardware management of attached services such as networks and storage.
- Automated instance management through APIs, command-line UI and WebUI for virtual instances.
- Access control for role-based access to individual hosts, such as GUI users, API user and administrators, amongst others.

Virtualization

This layer hosts the virtual machine manager, the Hypervisor. This allows the creation and maintenance of the virtual machines. Other virtualizations managed at this level include:

- Virtualizations for storage - Network attached storage (NAS) and Storage area networks (SAN).
- Network virtualization, creating private networks of VMs.
- Virtual hardware connectivity, providing UIs and APIs to allow services to be attached to the VM.

Operating System

This layer is the hardware management layer, representing the host operating system, packet routing for routing data on the network, and the device drivers managing the storage and I/O devices.

Hardware and Storage

The lowest layer in cloud computing stack is the physical (hardware) layer. This layer provides the following:

- Computation and host hardware control.
- Network connectivity.
- Short term memory.
- Long term disk storage.

3.2.3 Types of Cloud Models

Cloud computing is not only classified by the types of services provided, but also by the location these services are provided from. As seen in earlier sections, service providers are likely to provide various deployment models to their customers particularly with full SaaS still in a growth. These allow their customers to either host the application on-premises within their business, or as SaaS, where, as service providers they host the application suite and make it accessible over the internet. Mell and Glance (2011) define these cloud deployment models within three types: Public Cloud, Private Cloud and Hybrid Cloud.

Public Cloud

Within a public cloud, the infrastructure and services through which data is processed and stored is maintained off-premises, and accessed via the internet. The infrastructure is provided as a service by a cloud service provider and tailored for multi-tenancy. As it is a shared architecture, public clouds are cost-effective and efficient ways to share data with others. The “public” element of access however makes security an important consideration.

Private Cloud

In a private cloud, the computing infrastructure is dedicated to one customer and is not shared across organisations. In some ways, private clouds are not considered real examples of cloud computing. These private clouds tend to be more expensive and more secure when compared to public clouds.

Private clouds are of two types: On-premises private clouds and externally hosted private clouds. Externally hosted private clouds are also exclusively used by one organisation, but are hosted by a third-party specialising in cloud infrastructure. Externally hosted private clouds are generally cheaper than on-premises private clouds (Mastorakis, 2015).

Since these kinds of clouds can only be accessed by those given clearances by the business, data tends to be more secure against attacks.

Hybrid Cloud

Businesses could also decide to apply a hybrid cloud model, which involves both public and private clouds operating under one system. An example deployment could see data that needs to remain most secure on a private cloud, with the rest of the information on a public cloud. This allows the cloud system to adapt to the various business needs, but creates an overhead for the management of the cloud system, ensuring that the cohesive whole remains secure whilst still enabling communication to flow through.

The cloud model best applied to a business is very much dependent on the requirements at hand, for example whether there is a need for cost effective data access or high security of data, or even both. This flexibility must be weighed against the additional complexity introduced, for determining how applications are to be distributed across the public and private clouds.

3.2.4 Cloud Services Attributes

Cloud Services provide several new features over other computing paradigms, as detailed by Wang, et al. (2008) and Grossman (2009). The following is a brief description of some of them:

Scalability and on-demand services

Cloud computing provides resources and services for users on demand. The resources are spread over several datacentres providing unlimited storage capability and scalability.

Backup and disaster recovery

Comprehensive backup and recovery functions inbuilt within datacentres ensure services always available on demand.

User-centric interface

Cloud interfaces are location independent (and potentially device independent), allowing access by users through interfaces such as web services and web browsers, though dependent on availability of an internet connection.

Guaranteed Quality of Service (QoS)

The underlying selling point for the cloud model is the guaranteed Quality of Service (QoS) for users in terms of best possible infrastructure, such as computation performance, resource bandwidth, and storage capacity.

Autonomous system

Through virtualization, cloud services provide automations facilitating systems management, and making this transparently to end-users.

Flexible pricing and minimised cost

Cloud computing removes the need for capital investment, as suggested by Snowden (2010). Cloud services are paid for on demand and for the capacity required. Infrastructure is rented, not purchased allowing the costs to be controlled. Economies of scale enjoyed by cloud service providers further help reduce overall costs to end-consumers. This flexibly in pricing comes to the fore in lower costs of entry when a business is entering a new market, for example.

These services provide other advantages besides controlling storage and infrastructure costs, but also enable savings around the management of systems and applications, the requirement for hiring and training new staff, and even decreased on-premises utility costs.

Reduce run time and response time

Applications that need to offer a fast response time to their customers can use cloud services such that any computational intensive tasks are farmed out to virtual machines optimising response time, scaling on demand.

Minimize infrastructure risk

Risks inherent in the procurement of physical servers are minimised through cloud services. Issues such as the required scale of infrastructure become the remit of the cloud service provider. This infrastructure risk is also minimised by enabling surge computing, allowing resource supply to meet demand spikes.

Increased pace of innovation

Lower cost of entry to new markets allows start-up companies to deploy new products quickly and at a low cost through a cloud model. Small businesses can therefore compete more effectively against traditional organisations whose enterprise models result in longer deployment processes.

3.3 Summary

Comparison of these six computing paradigms resulted in a view that cloud computing has conceptual parallels to the other computing paradigms. There were several important differences across the paradigms however. This chapter detailed how mainframe computing offered finite computing resources, whilst the fundamental concept of cloud computing is that power and capacity can be scaled indefinitely to meet the requirements of the service at a specific point in time.

Differences also exist in how mainframe computing resources were consumed through dummy terminals which acted as user interface devices. Cloud computing on the other hand also supports access through powerful local personal computers, which could be providing processing support within specific service models.

Cloud computing was then analysed in detail, with a review of cloud architecture, the various types of cloud deployment models, and the key attributes of a cloud service. These concepts were then charted against the evolution of ERP, detailed in the next chapter. This enabled the merging of the two research elements together to propose a forward-looking view of possible future architecture models, as described in chapter 9.

CHAPTER 4

EVOLUTION OF ENTERPRISE RESOURCE PLANNING (ERP)

4.1 Introduction

This chapter details the evolution of ERP systems from basic principles to the complex business systems available today. It describes how the evolution of the technology covers both functional elements of the systems as well as their underlying architectures. The computing paradigms from Chapter 3 are applied comparatively over the ERP timeline, to highlight the close relation evident between complex business systems, and the underlying computer architectures supporting them. This enables the extrapolation of a forward view as to how future ERP systems could be structured architecturally to support more complex business cloud services.

4.2 ERP Timeline

ERP systems are described by Rashid, et al. (2002), as software applications for business management, designed to improve the performance of an organisation's resource planning and operational control. ERP software is generally modular in structure, integrating activities across functional departments, and supporting a range of functional areas such as planning, manufacturing, sales, marketing, distribution, accounting, financial, human resource management, project management, inventory management, service and maintenance, transportation and e-commerce. Leon (2014) stated that *"having an ERP system is not a luxury but a necessity. It is a must for survival in this competitive world."*.

ERP evolved from Manufacturing Requirements Planning (MRP) II, which on the business-side expanded on the coordination of manufacturing processes to the integration of enterprise-wide backend processes.

The ERP software architecture provides a flow of information between the functions of an organisation in a consistent manner. As defined by Davenport (1998), this allows the replacement of standalone legacy information systems, which handled specific parts of a business and were not easy to integrate into one seamless control system. Table 2 summarises the functional evolution of ERP systems from the 1960s to 2000s.

Table 2: ERP Evolution Timeline (adapted from Zhang and Sysoptima, 2005)

Timeline	**System**	**Description**
1960s	Inventory Management & Control	Inventory Management and control is the combination of information technology and business processes of maintaining the appropriate level of stock in a warehouse. The activities of inventory management include identifying inventory requirements, setting targets, providing replenishment techniques and options, monitoring item usages, reconciling the inventory balances, and reporting inventory status.
1970s	Material Requirement Planning (MRP)	MRP utilises software applications for scheduling production processes. MRP generates schedules for the operations and raw material purchases based on the production requirements of finished goods, the structure of the production system, the current inventories levels and the lot-sizing procedure for each operation.
1980s	Manufacturing Requirements Planning (MRP II)	MRP II utilises software applications for coordinating manufacturing processes, from product planning, parts purchasing, inventory control to product distribution.
1990s	Enterprise Resource Planning (ERP)	ERP uses multi-module application software for improving the performance of the internal business processes. ERP systems often integrate business activities across functional departments, from product planning, parts purchasing, inventory control, product distribution,

		fulfilment, to order tracking. ERP systems may include application modules for supporting marketing, finance, accounting and human resources. Examples being SAP, Axapta, Navision.
2000s	Web enabled Enterprise Resource Planning (ERPII)	ERP applications become more functionally comprehensive and sophisticated. Increasingly being adopted not only by larger Tier 1 organisations, but also within small and medium sized businesses. This resulted in higher market share adoption year on year. Consolidation of ERP solutions within the market has seen bigger industry players backing up research and development (R&D) in ERP systems, with ever larger budgets. Elements of functionality and reporting become accessible through the web, and increasing use of web services for integration.

It should be noted that ERP solutions are generally classified in Tiers, from 1 to 3 (Osintsev, 2016). Tier 1 in this case refers to the most complex and expensive solutions, generally aimed for enterprise level businesses. At the other end of the spectrum, Tier 3 are the simpler lower cost solutions, such as Sage50, targeted at smaller commercial entities. An overview of these tiers is presented in Table 3: ERP Tier. Osintsev (2016) further proposes that cloud computing has blurred the lines across these tiers, therefore making these classifications less significant.

Table 3: ERP Tier Classes (adapted from Osintev, 2016)

Vendor Tier	Complexity	Business Size	Revenues	Time to deploy	Scalability	No. of Users	Cost of ownership
Tier 1	High	Enterprise / Global	>$100M	Long	High	>250	High
Tier 2	Medium	Small and Medium	$1M - $100M	Medium to Long	Limited	5 to 250	Medium
Tier 3	Low	Small	<$1M	Short	Low	1 to 50	Low

4.3 ERP Architecture

4.3.1 Two-Tier Architecture

Organisations have been transitioning from systems developed in-house, and third-party solutions using differing and incompatible platforms; to integrated ERP systems (Rashid, et al, 2002). Integrated ERP solutions make better use of system architectures that support the resultant increased transaction and data storage requirements, whilst still ensuring a consistent solution a business could operate on.

For these reasons, ERP system architectures followed closely the key computer architectures of their period. A common legacy ERP architecture was the two-tier client server model. This allowed the creation of a decentralised business computing environment within which multiple end users could work, with business transactions updating a central database, as shown in Figure 15.

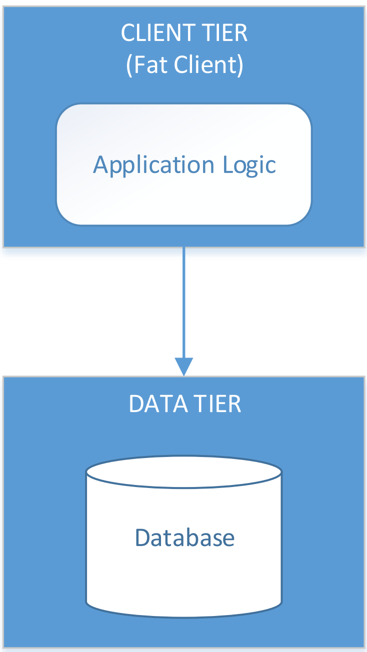


Figure 15: Two-Tier Architecture - Fat client

With increases in network bandwidth, this allowed the two-tier, client-server model to support a thin-client tier architecture, with the application and business logic handled by the server within the data tier. This enabled centralised management of the ERP system whilst still providing a de-centralised use of the applications.

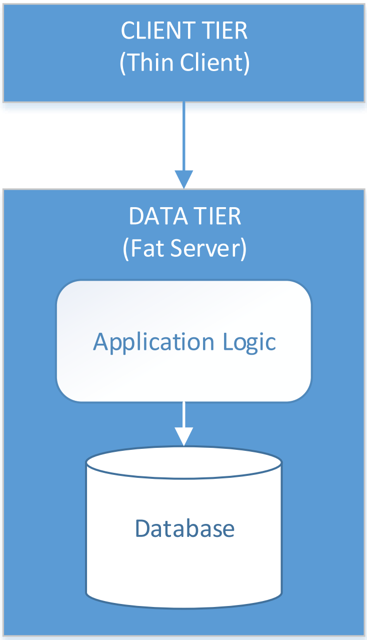


Figure 16: Two-Tier Architecture - Thin Client

4.3.2 Three-Tier Architecture

The two-tier architecture posed some limitations on scalability and flexibility, since as a business experienced growth and complexity, this tended to outgrow the solution itself, as described by Exforsys (2007). Additionally, the limited resources of both clients and server hardware could have an impact on system use experience. This led to the evolution of the three-tier architecture. The three-tier architecture is still a client-server architecture within which the functional process logic, data access and user interface are developed and maintained as independent modules on separate platforms.

This ERP architecture allowed for the managing of data independently from the physical storage, facilitating migration to new environments. As described by Marston (2012), the client does not have direct access to the database, which results in additional security for the business logic. This security also means that if one of the tier fails there would be no data loss.

The World Heritage Encyclopedia (2016) describes how at the application layer level, middleware was developed allowing various software components to be seamlessly accessed supporting more complex, distributed solutions and leveraging other emerging technologies within the computer industry.

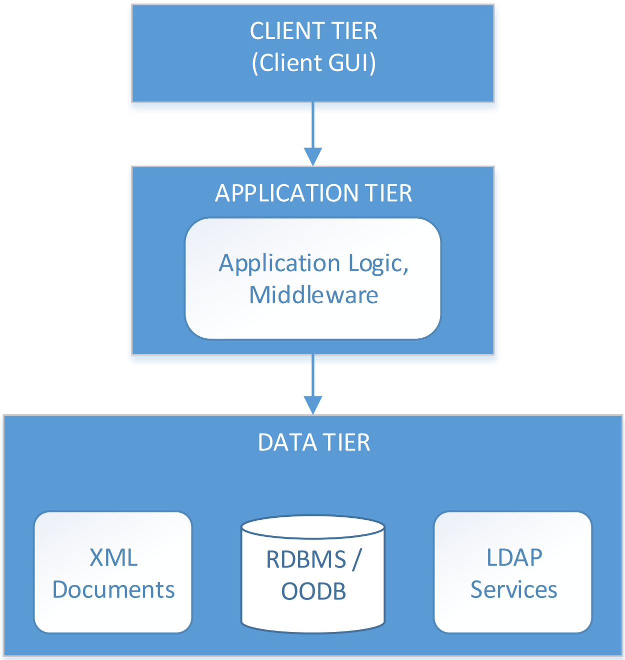


Figure 17: Three-Tier Architecture

4.3.3 Multi-Tier Architecture

The multi-tier, multi-layer architecture further extends upon the three-tier architecture described, through additional physical and logical separation of the client, presentation, application and data services tiers. Hill, et al. (2009) define each of these tiers as further subdivided into other functional and service layers. These elements would each be running on a separate server, or separate clusters that can each provide the services at top capacity, mitigating the pitfalls of resource sharing. This separation enables managing each element of the ERP system easier, facilitating the isolation of problems should these occur and ensuring data integrity in instances of failure within the elements. Multi-tier architecture is also referred to as N-tier by the computing industry, to imply any number of tiers.

This ERP architecture provides the following advantages to an ERP system:

- **Scalable** – Businesses can scale separate tiers without impacting other tiers.
- **Individual management** – Prevents cascade effects in case of failure, and isolates maintenance.
- **Flexible** – Facilitates expansion both horizontally and vertically per requirements.
- **Secure** – Each tier can be secured separately, abstracting direct access to database.

It should be noted that there is distinction between multi-tier and multi-layer. The former implies the physical architecture of the system, whilst the latter refers to the logic structure of the system. As Figure 18 shows, a system could be both multi-tier and multi-layer.

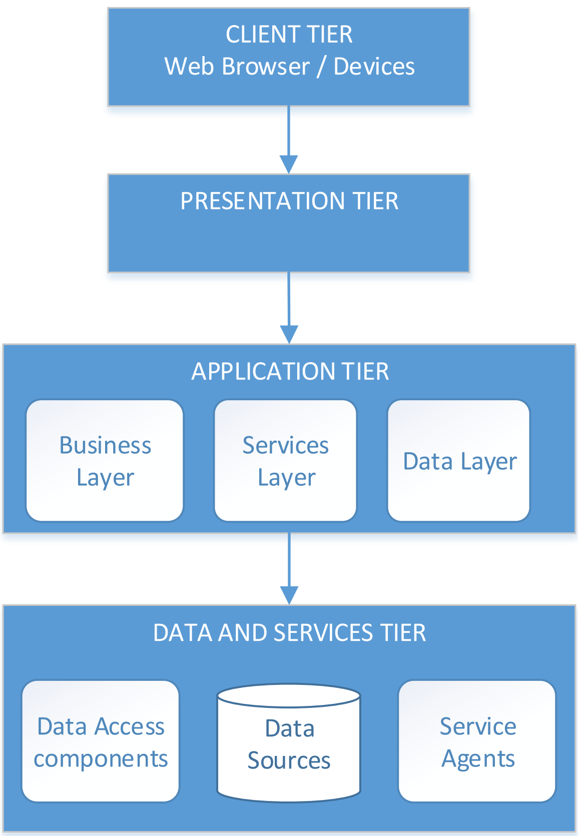


Figure 18: Multi-Tier / Multi-Layer Architecture

In addition to the advantages that result from this architecture, a more structured approach to software engineering is enforced. Best practice procedures need to be followed to ensure that each layer still works seamlessly across the ERP system. As highlighted by Marston (2012), some such rules are:

- The code for each layer must be contained with separate files which can be maintained separately.
- Each tier may only contain code that belongs to that layer. Thus, business logic can only reside in the Business layer, presentation logic in the Presentation layer, and data access logic in the Data Access layer.
- The Presentation tier can only receive requests from, and return responses to, an outside agent. This is usually a user, but may be another piece of software.
- The Presentation layer can only send requests to, and receive responses from, the Business layer. It cannot have direct access to either the database or the Data Access layer.
- The Business layer can only receive requests from, and return response to, the Presentation layer.
- The Business layer can only send requests to, and receive responses from, the Data Access layer. It cannot access the database directly.
- The Data Access layer can only receive requests from, and return responses to, the Business layer. It cannot issue requests to anything other than the database management system (DBMS) which it supports.
- Each layer should be isolated from the inner workings of the other layers. The Business layer, for example, must be completely database-independent, with no direct interface with the inner workings of the Data Access object. It should also be presentation-agnostic and not know or care how its data will be handled. It should process its data consistently, irrespective of what the receiving component will do with that data. The presentation layer may take the data and construct an HTML document, a PDF document, a CSV file, or process it in some other way, but that should be totally irrelevant to the Business layer.

4.4 Representative ERP Systems

Microsoft Dynamics AX and NAV are referenced here as representative examples of proprietary ERP software systems. Both these systems could support complex business requirements within a range of industries, and have been deployed globally across a large number of organisations. On-going R&D investment in these systems also means continuous evolution of the technology, with newer versions being released periodically to make best use of architectural advances. Their selection as reference systems within the current research was due to these characteristics, as well as a result of ease of access through the author's professional background.

4.4.1 Dynamics AX

Dynamics AX is Microsoft's flagship ERP solution, and a leading tier-1 system in the industry. It covers functionality for running core business requirements such as financial management, human resources management, manufacturing, supply chain management, procurement and sourcing, project management, sales and marketing, services, retail, business intelligence and reporting.

Gartner (2011) details how AX architecture allows for a flexible, customisable solution. It is easy to customise and integrate, and can be modified to suit a variety of applications. It is also based on standard Microsoft technologies, which facilitates development and customisation. It is a very versatile solution that can be used for a variety of industries, including manufacturing, distribution, retail, professional services and the public sector. Additionally, this solution can support companies who need multi-location, multi-currency and multi-language facilities.

AX architecture offers an open source code environment, having an N-tier architecture with no limit on number of users. An Enterprise Portal provides SharePoint web authoring functions together with document management. This in-depth functionality allows the system to meet industry needs in line with other main competing ERP systems like Oracle and SAP.

Usability is also key, with the user interface being very customisable, having role-based personalisation, and has consistent look and feel to other Microsoft software.

4.4.2 Dynamics NAV

Dynamics NAV is also a Microsoft ERP software suite, primarily aimed for mid-sized to large organisations. The system offers specialised functionality for manufacturing, distribution, government, retail, and other industries.

NAV offers applications for financial management, human resources management, manufacturing, multiple and international sites, project management, sales and marketing, service management, supply chain management and business intelligence. The functionality is particularly elaborate in manufacturing and distribution. The system is known for being highly customisable and partners have developed a long list of industry-specific configurations to serve various vertical markets, such as the one approached for the case studies in Section 7.4.1.

The system adapts to the requirements of various regions of operation, even for highly specialised industries and organisations. The architecture allows for ease of use and offers rapid adaptability and simplified customisation, enabling complex businesses to easily add functionality, custom applications and online business capabilities. These capabilities are structured within one integrated system.

4.5 Summary

This chapter has examined how ERP systems have developed over the years, together with their underlying ERP architecture models. These models were evaluated in parallel to the computing paradigms detailed in Chapter 3, with consideration for the cloud services architecture detailed earlier. The Dynamics AX and NAV solutions were described, referenced as representatives of ERP systems able to support the requirements of complex organisations.

Based on this analysis the author extrapolated upon these architectural trends in later chapters, proposing a view on the architectural model and strategies required to support complex ERP systems within cloud service models.

CHAPTER 5

A NEW FRAMEWORK FOR COMPLEX BUSINESS SYSTEMS

5.1 Introduction

As described in previous chapters, business solutions have been evolving on both a technical and functional level, increasingly extending services into the Cloud. These solutions have become characterised by increasing complexity of deployable components, making it increasingly difficult for businesses to make informed decisions on the best solution options available to them. Traditionally this process would have been supported by a solution provider, through defined methodologies involving in-depth analysis of the business within the context of an implementation project. The cloud model however changes this, with the customer required to be more proactive within the procurement process, even before selection of a service provider or solution. This research has therefore hypothesised the requirement for a framework (Step 3 in the methodology), to support and empower the business in this strategic decision-making process, providing the tools facilitating transition into a cloud hosted service model.

This chapter describes the components constituting the SaaS Complexity Framework (SCF), and explains the importance of understanding and quantifying complexity within business. Existing models for complexity, covering both business and systems, are examined within the context of this framework. The outcome of this analyses resulted in the requirement for an alternative definition of business complexity, as well as the development of a related metric for measuring this complexity. This chapter also explains how this new complexity metric can be validated through the application of existing metric validation models, and describes the importance of this within the wider framework development.

5.2 Measuring Business Complexity

As outlined in the research chapters, there are close links between business complexity and system architectures, with one influencing the other over the years. Accordingly, this research concludes that for a business, proper understanding of its own complexity is critical to enable the best technology decisions in the strategic procurement of their IT solutions. This also serves to identify the ideal technological architecture required to support that complexity. This requirement to measure business complexity defined the second step in the research methodology.

As new computing paradigms are developed with cloud computing, evolving system architectures provide increasing ERP solution complexity, and a multitude of deployment permutations. The author observed however that a disparity exists between current SaaS systems and complex business requirements. A SaaS solution would for example provide a product procurement process to enable a business to purchase goods to stock, however the physical business might have a more complex procurement process setup with cross-border goods sourcing, covered by currency hedging functions. At a high-level therefore, the SaaS solution seems to meet requirements by providing procurement functionality, however in the more complex business requirements, the disparity becomes evident. The requirement for a new framework was identified as a means for identifying and reducing this disconnect.

The SCF framework aims to provide organisations with the means to quantitatively analyse their business structure and their complexity requirements; evaluating these against comparable solutions and service providers. This new framework is also intended to apply to cloud service providers, when evaluating an industry or market to provide services to. Figure 19 describes the structure of the SCF framework, together with its related components. It also provides examples of potential real-life commercial applications for it.

The SCF framework is comprised of three interrelated components:

- *SCF Unified Model (SCFUM)* – This model represents a unified view of various business processes and functions, common across industries, which serves to model a business, industry or solution within the framework for analysis. This is detailed in Chapter 6.

- *SaaS Complexity Metric (SCM)* – The complexity metric quantifies the measure of complexity within a business, industry or solution, and expresses this in a standard unit (SCI). The mapped SCFUM is used for the calculations. This is detailed further in this Chapter 5.
- *SaaS Complexity Index (SCI)* – The measure of complexity calculated though the SCF is represented in units of SCI. An application of this is detailed in Chapter 7.

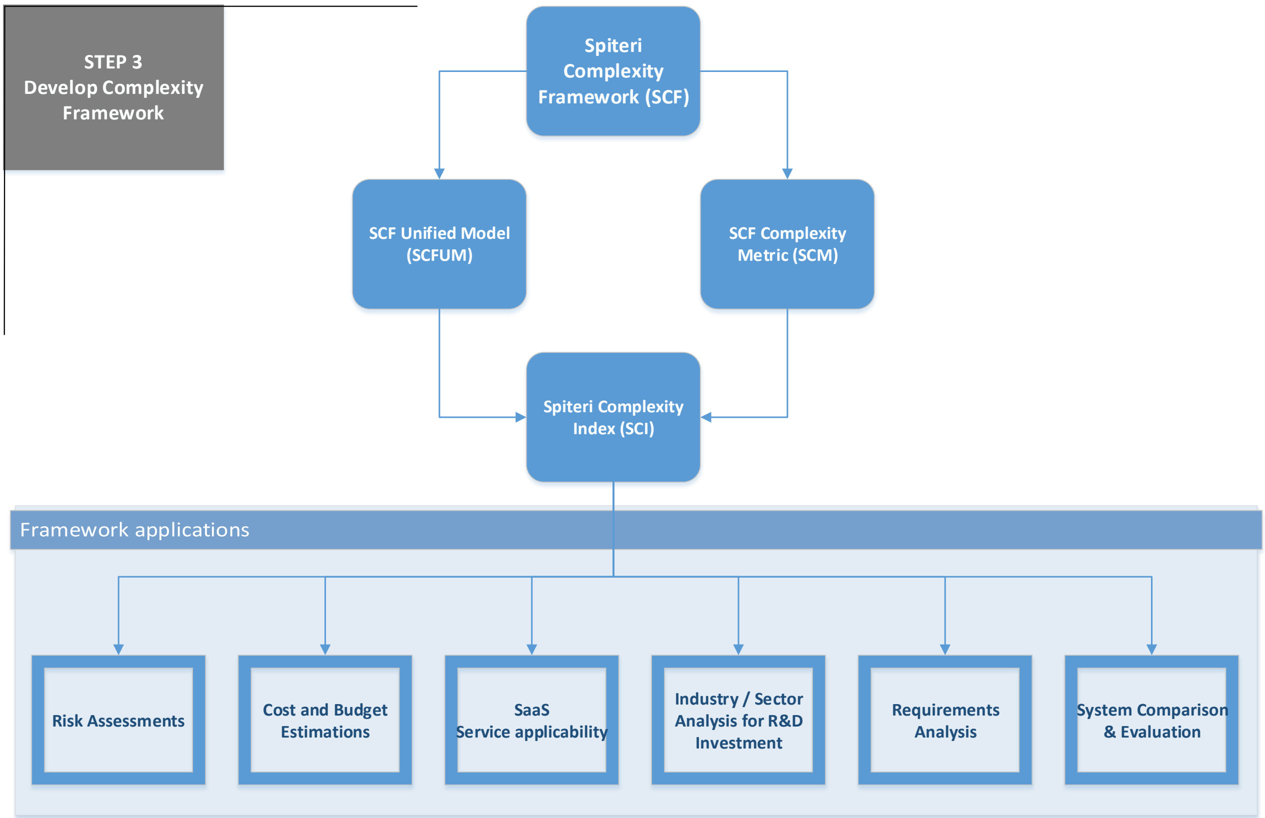


Figure 19: Complexity Framework Structure

5.2.1 Complexity in Business and ERP Systems

The definition of complexity varies widely, depending on the science that it is being applied to. Some examples would be algorithm complexity defined in computer science, logistical complexity within operations management and organism complexity within biological sciences. As researchers Amaral and Uzzi (2007) observed, there is evidence of complexity in many of these disparate systems, where it was also noted that on simplifying complexity concepts, generic aspects of these apply across a wide-spectrum of research fields. This in turn has led to an understanding, both qualitative and quantitative, of the complex systems encountered in nature, technology and everyday systems.

Latva-Koivisto (2001) noted the importance of finding a complexity measure for business process models. Hoppermann (2010) describes business complexity as being the condition of having interconnected and interdependent stakeholders such as employees, customers, partners, investors and competitors, amongst others, as represented in Figure 20. These definitions of business complexity were found to have a wider scope, covering the business landscape, whilst excluding the underlying supporting systems and technology elements. Consequentially, these were deemed not to be directly applicable to the requirements of the SCF framework. The consideration here is that a business would only have direct control on its own processes, structures and underlying systems, and accordingly these are the elements the SCF framework is applicable to. This is represented by the central business circle within the same Figure 20.

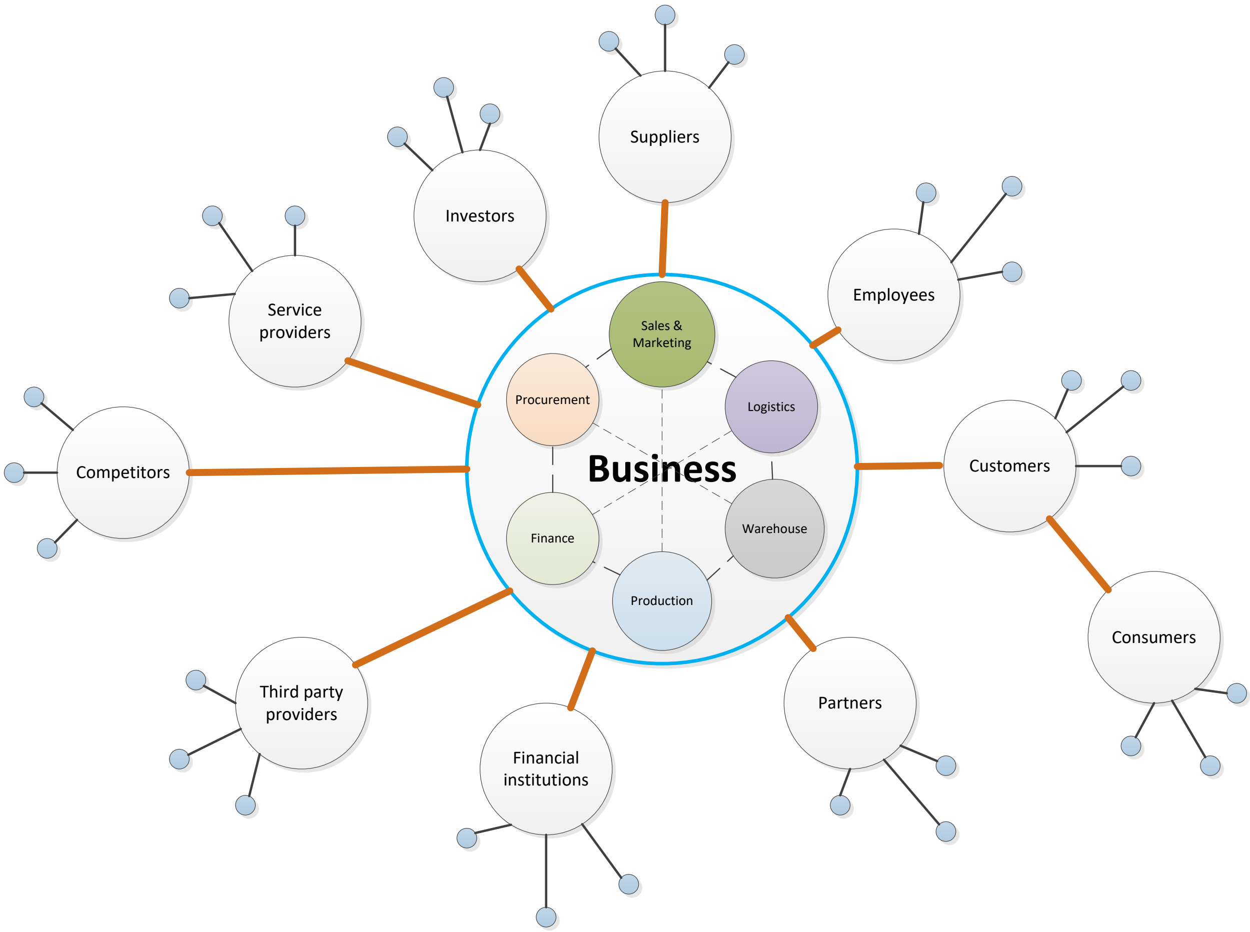


Figure 20: Complexity as a Condition of Interdependent Stakeholders (Spiteri, 2012)

Researchers have also observed that there are many similarities between software programs and business processes (Ghani et al., 2008; Vanderfeesten et al., 2007; Cardoso et al., 2006). The notion that processes can be treated like software was first suggested by Osterweil (1997), who argued that technologies that are used to build application software can also be used to build processes, as the two show structural similarities.

The current research reviewed various models of complexity, across both business processes and system processes. These formed the basis from which the author could derive a new definition of complexity for business and underlying systems. The notions by Skyttner (2001), that it is essential to view a system from a wide perspective and understanding the whole structure as well as its interdependencies, were here considered. The author has applied three key theories of complexity; Complex system theory (Section 5.2.1.1), Programmatic complexity (Section 5.2.1.2) and Network complexity (Section 5.2.1.3), to business processes and ERP systems. The aim being that of defining a complexity model against which the SCF framework could be validated. An overview of these concepts was published in the journal of internet technology and secured transactions (JITST), Spiteri (2012b). Figure 21 represents the kind of overlap across business and system complexity for which a new cohesive definition was required.

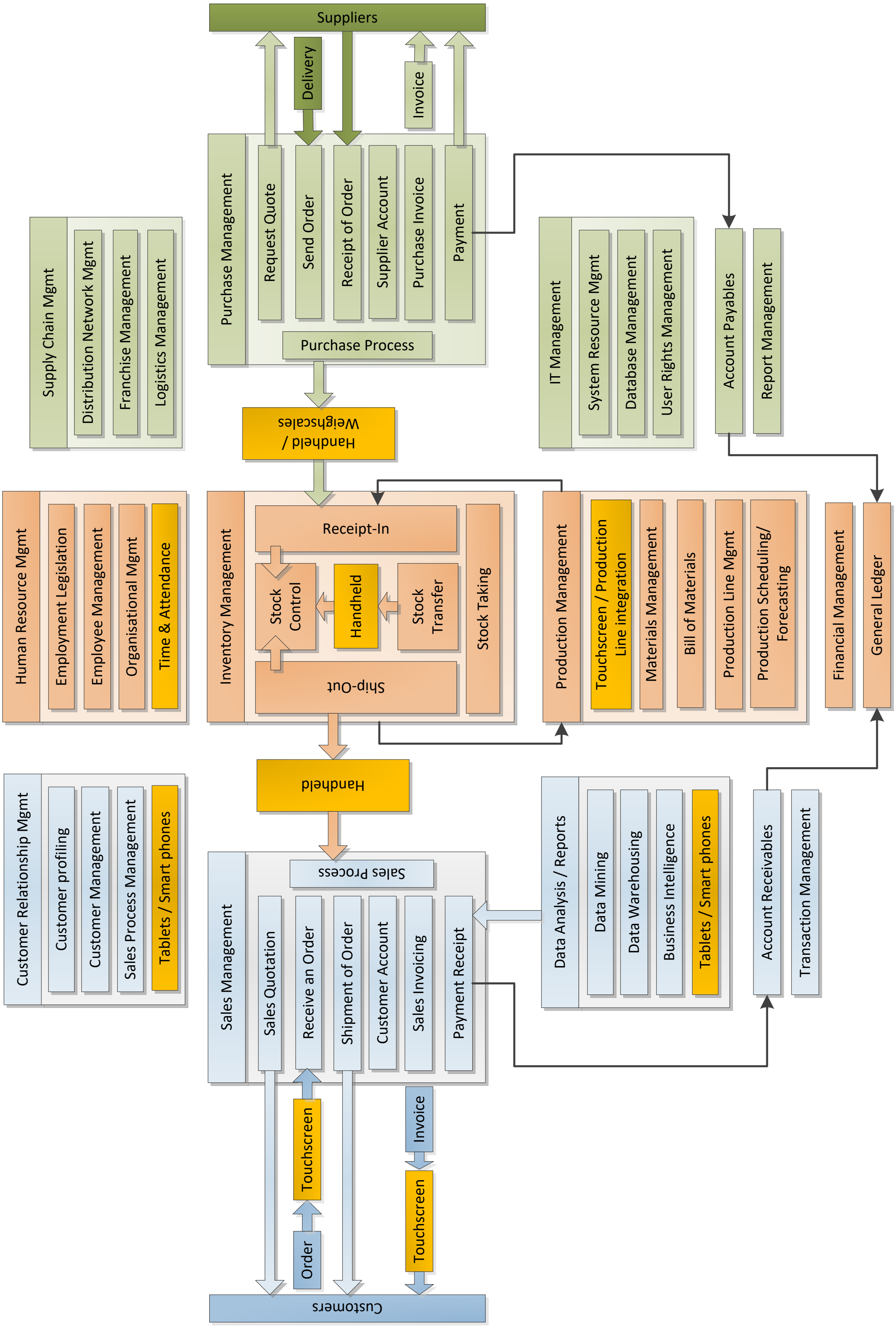


Figure 21: A Schematic of Process Complexity and Interconnection (Spiteri, 2014)

5.2.1.1 *Complex Systems Theory*

Complexity theory is the study of complex, nonlinear, dynamic systems where the resultant output is of a simple nature (Levy, 2000). This is the distinguishing factor from chaos theory, of which primary concern is with systems in which the recursive application of nonlinear deterministic functions can give rise to apparent random behaviour and subtle patterns. In this case, a simple system can result in complexity in its output, and as such the theory is focused more on the output than on the system itself.

Complexity theory's systems approach led to the creation of a variety of definitions for complex systems. These systems are described as complex because they have numerous internal elements, and dynamic because their global behaviour is governed by local interactions between their elements (Geyer et al., 2005; Richardson, 2004).

Organised vs. Disorganised Complexity

The discovery of common abstractions and mechanisms greatly facilitates the understanding of complex systems (Edmonds, 1999; Young et al, 2007). For example, in an organisation where a service is provided, a product is manufactured, material is purchased, wages are paid, and statutory financial details are reported; recognising the properties common to all such organisations allows more easily to validate concepts across such entities.

Researchers have noted that systems do not express a single hierarchy, but rather various hierarchies are usually present within the same complex system (Simon, 1973; Richardson, 2004). The author's view is that this concept can be applied to a business, for example by breaking down an organisation into its various hierarchies, such as its Purchasing department, Production department or Finance department. This decomposition of the whole represents a structural hierarchy, which is part of a wider hierarchy. According to Skyttner (2001), it is essential to view a system from both perspectives, understanding the whole hierarchy structure as well as its sub-hierarchies.

An organised complexity model proposes a view that there exists a hierarchy of levels of organisation, with each level more complex than the underlying one. A level will be representative of emerging elements which do not exist at the lower levels. The author extends this concept to the ERP system model, where therefore, as an example, within a

Purchase Order Processing hierarchy, a Product Receipt emerges from Purchase Document Lines indicating Product, Units and Prices. These in turn emerge from a Purchase Document header which indicate Supplier details.

Analysing a complex software system such as an ERP, leads to the identification of multiple elements that interact in a multitude of intricate ways, that is, complex. In some instances, there is little definite commonality among either the parts or their interactions evident. This is then considered an example of disorganised complexity (Checkland, 1981). The author proposes that in modelling ERP business complexity, the organised elements as well as the disorganised ones, must be considered together within one whole.

Tangible vs. Intangible Complexity

In addition, the author's model of complexity, as applied to an ERP system, was found to have to consider the interface elements within the operational environment of the system for more accurate modelling, since this could have a direct impact on the overall complexity of the system and therefore the required framework. To this end, the research factored complexities that can be differentiated as those of a tangible and of an intangible nature.

Tangible complexity can be identified as industry complexity, regulatory complexity, organisational complexity, and process complexity, which in a system could take the forms of reports and interfaces. The connection between the entities in Figure 20 provide a representation of this.

Intangible complexities would relate to scenarios such as Monte-Carlo simulations, unintended consequences, correction of errors, sustainability, system expansion potential and maintenance management. These could provide weighting factors to the framework.

As organisations experience growth, the tendency for complexity within the business increases, as discussed in Section 5.2.1. This frequently results from management decisions to adapt to internal and external pressures, and through externalities or market forces (Kimberling, 2010; Jagersma, 2004). It is quite likely this complexity would in turn translate into the IT systems deployed to manage the organisation itself. Highly complex processes are error-prone, difficult to understand and difficult to maintain (Muketha et al, 2010), and this is reflected in the systems implemented to manage them.

5.2.1.2 *Programmatic Complexity*

Since an ERP system is inherently a software system, the author also had to consider the impact of programmatic complexity that the ERP model would be defined by. Programmatic complexity is the numerous elements defining the software and their interactions. As the number of elements increases, the interactions between them are seen to increase exponentially (Kearney et al, 1986; Megiddo, 1987). These interactions could flow within any direction of the element, within or without a hierarchy structure.

When considering data access complexity, as an element within the hierarchy receives a message, it acts upon it as defined by functions inside that element (Card and Agresti, 1988). As elements are arranged together within a business process or system process reflecting it, then the definition of its communication access points allows other elements to be exposed and to send them messages.

Dataflow complexity, another subdivision of Programmatic complexity, is defined by the functional elements within the software hierarchy connected by directed dataflow queues (Falgout, 2011; McCabe, 1976). The dataflow queues, transport the data between connected functions, with the output of one functional element being the input of another. The author observed that within an ERP process, this could translate to distinct functions manipulating data within the hierarchy levels. In this case, the consolidated data inputs and outputs result in the general inputs and outputs of the hierarchical levels and elements making up the process nodes, as shown in Figure 22.

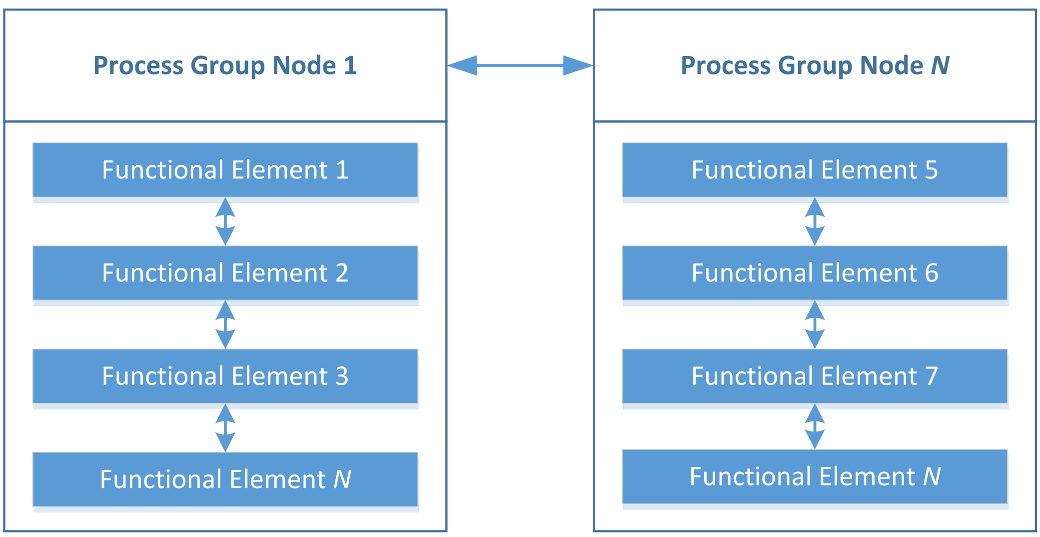


Figure 22: Process Nodes and Functional Elements

5.2.1.3 *Network Complexity*

The research has applied notions from the models of Systems theory and Programmatic complexity to an ERP system model. The concepts of interactions between hierarchies and elements were found to be described through interrelationships of functional nodes within a network, where the state of each node is the function of its connections to other nodes.

The author then concluded that Network complexity theory had applications in defining this behavioural model within the development of the SCF framework. With Network theory's focus on emergent order and patterns in complex systems, as suggested by Levy (2000) and Wolfram (1985). The SCF framework took from the network model's essence of capturing the interaction among the many functional elements in a system, by modelling large numbers of nodes connected by simple logical rules. Figure 23 shows a representation of the concept as applied to the SCF framework, extending upon the notions described in Section 5.2.1.2.

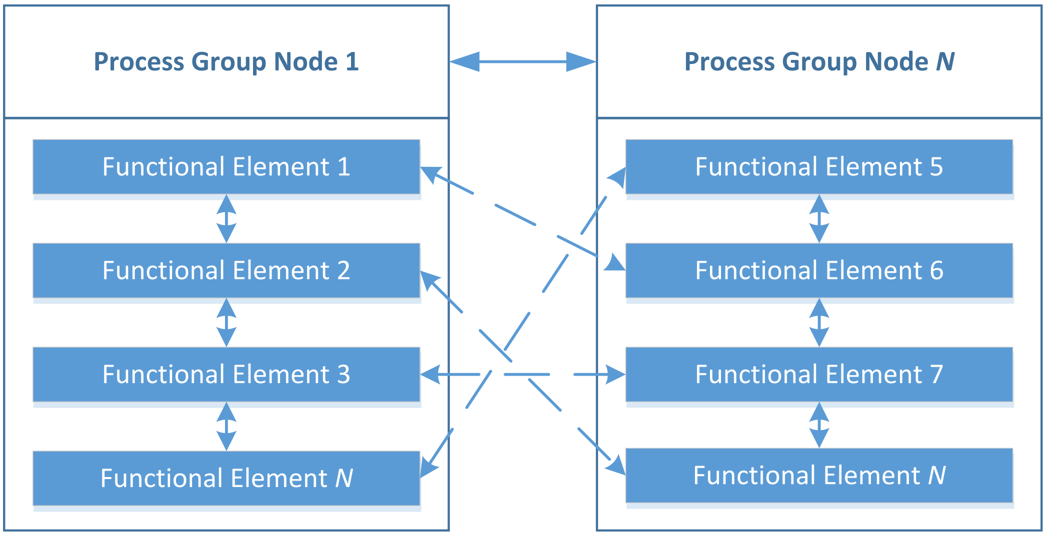


Figure 23: Process Functional Elements Interactions

An advantage in cross-mapping such complexity models to this business framework is that it allows use of the mathematical concepts available to describe such systems. One such example would be the "Bailey-Grossman Equation", $B(N) = \text{Max}\, j,\ X[j] \geq j$.

The "Bailey-Grossman equation," provides a single number, the Network Complexity Index (NCI), which is the balance point between the number of groups that are interacting

(X[j] ≥ j) and the size of those groups (Max j). This provides a means to manage the growth of a network's complexity through the ability to measure the rate of that growth, by quantifying the complexity level at any given moment in time. (Bailey and Grossman, 2012).

This potentially provides the means to calculate possible network complexity of future business systems by applying this formula within an SCF framework application, providing another dimension of complexity within a decision process. This is however outside the scope of this research, and could be considered for future work.

5.3 Defining Business Complexity for Business Systems

As discussed in Section 5.2.1.1, for the complexity theories and models described, a common definition can be inferred that complexity is the expression of numerous elements in a system where these elements are interrelated (Lucas, 2006; Amaral and Uzzi, 2007). This contrasts with simple systems, which would therefore have a small number of easily defined elements. In this study's application of these theories to business and ERP model, the author defines baseline business complexity as a measure of tangible complexity, this being the composite of business and technological elements determined by factors such as the number of sub-processes as well as their interactions, interdependencies and relationships within the process environment (Spiteri, 2014). When these features are applied to a business, the author's definition covers organisation structure and interdependence of business process elements, such as:

Finance - for book-keeping and financial transaction recording, enabling audit and the creation of management accounting reports.

Sales Order Processing - which facilitates and records sales of product or services and related transactions, including related document transactions and reporting.

Purchase Order Processing - which facilitates and records purchases of product or services and related transactions, including related document printouts and reporting.

Warehouse and Inventory Management - to ensure accurate stock management, which is usually a key element for accurate business flow within other elements of the ERP, enabling more efficient use of product inventories, control of picking locations, shelf live and storage.

Manufacturing - allows production companies to manage their bill of materials, production process routings, production scheduling and recording product consumption and output; including waste and production costs.

Human Resources - facilitates management of employee data, payment of skills quotients, salaries, vacation and absence records.

Services Management – allows service companies to manage the provision of their services to customers including the handling of warranty claims.

Customer Relationship Management – allows management and analysis of customer interactions and data throughout the customer lifecycle, with the goal of improving business relationships with customers, yielding data relevant to the marketing teams.

Transport Logistics – management of the business transport requirements in timely distribution of product from suppliers to customers, both for businesses owning their own fleets or third-party transport. This allows for more efficient planning and control of carbon costs of transportation directly within the system business process.

Quality Control – Management and recording of quality assurance and controls for product and related processes ensuring that the business facilities are operating within appropriate quality conditions and ensuring fitness for purpose of the product.

Sales and Purchase Forecasting – as a tool for planning both current and future, supply and demand, for the product.

The context of these elements within a business process is exemplified in Figure 24, which shows the interactions of the functional elements within the Order to cash and Return to credit process node. From the author’s professional experience, spanning several years of business analysis and solutions implementation, such business functional elements and their processes have been consistently evidenced within a wide range of organisations. These have also been found represented within ERP systems as standard modules integrated within the application.

The author proposes that the more interrelated processes and services an organisation has, the more likely that complexity will be present. Thus, causing issues to the integration of SaaS ERP within its business structures. A representation of such business complexity through interrelated processes is shown in Figure 21.

This alternative definition of complexity, which expands on the general conditions of complexity in business suggested by Latva-Koivisto (2001) and Hopermann (2010), can be surmised as follows:

Author's definition of business complexity: *Business complexity is evidenced as an elevated number of business processes operating interdependently within system functions, with such a system falling under the categorisation of what could be considered a fully-fledged, ERP system implementation.*

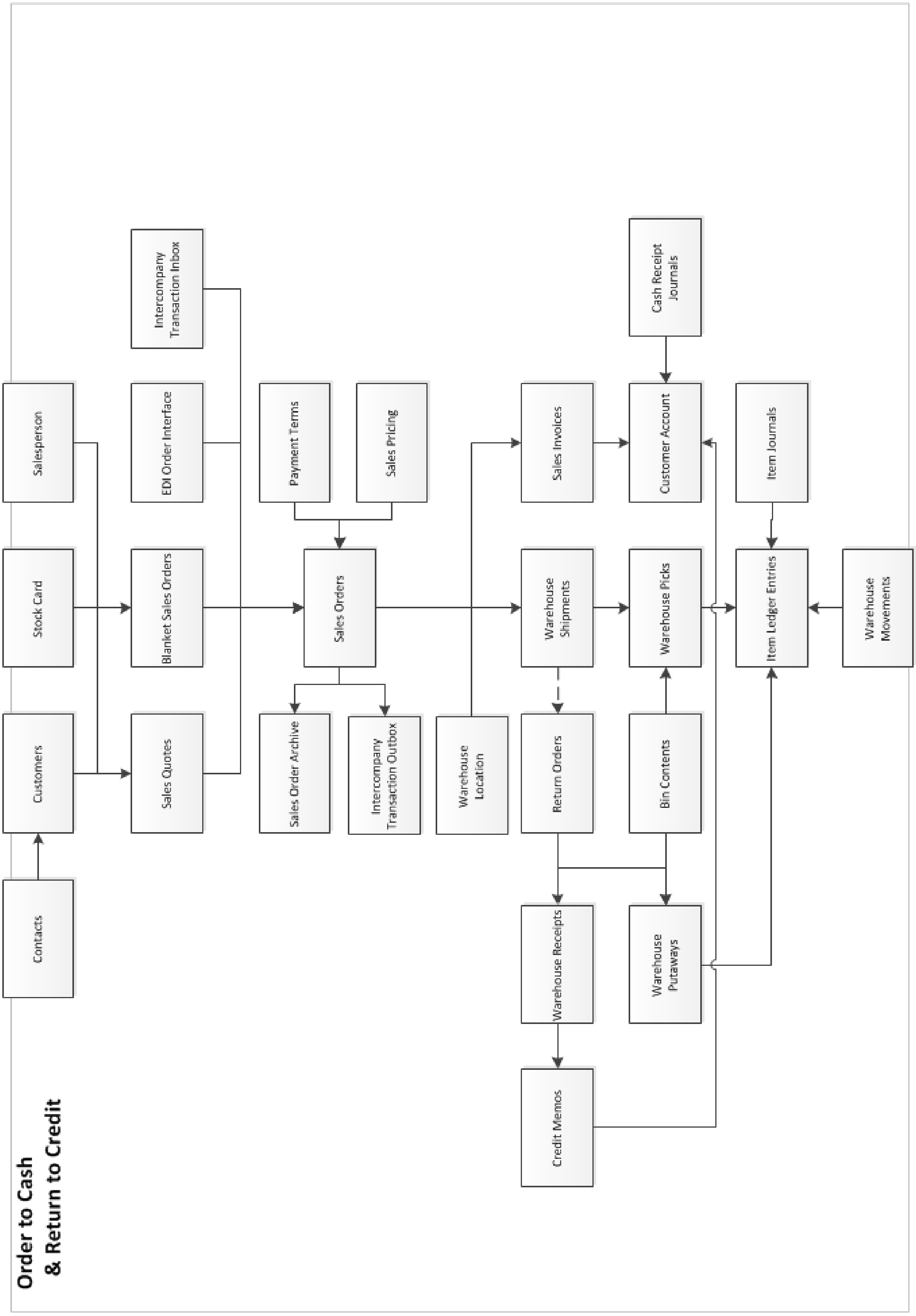


Figure 24: Example Sub-Processes within the Complex System (Spiteri, 2012)

The research discovered that an ERP implementation would not necessarily imply a complex business system by default however, since partial modules can be implemented separately to cover only particular aspects of the business. Therefore, applying the Spiteri (2012a) definition of business complexity, a complex ERP system deployment would be one that demonstrates that most of these functional elements are in operation within the business environment. This has been modelled in this work through business case study calibration.

As a result, the author also surmises that business complexity would increase further if disparate systems, technologies and devices were used to handle the various business elements, as this would traditionally require data interfaces, mapping structures and communication protocols, with some of these potentially being provided as services through third parties having their own proprietary systems (Spiteri, 2014). This is where other complexity models, as explained in Section 5.2.1.3, could have a role to play.

5.4 Defining a Measurement Metric

Measurement is key to many systems, particularly for companies and trading enterprises. This allows to both define such systems, and to understand and trust them. There are varied types of measurements with a variety of uses, however each of these has a common theme, some aspects of an entity are assigned a descriptor, which enables them to be compared with other elements. According to Fenton, et al. (1997), the definition of measurement is therefore that of a process by which numbers or symbols are assigned to attributes of entities in the real world in such a way to describe them according to clearly defined rules. A measurement is constructed by two elements:

- An **Entity** – this being an object in the real world.
- An **Attribute** – this being a feature or a property of that entity.

Attributes are often defined using numbers which serve as abstractions to reflect a perception of the real world. This enables concepts to become more visible and therefore more controllable. Science defines two kinds of quantification:

- **Measurement** - this being a direct quantification as in measuring the length of something.
- **Calculation** – this is indirect combination of measurements into a quantified item.

DeMarco (1982) suggested that "you cannot control what you cannot measure". A measure however needs to be associated with a model to provide a representation of how the measure maps the attributes and entities in the real world to the elements of a numerical system.

The process of defining new metrics can therefore be surmised as involving four main steps which follow each other in sequence:

i) determine category of entity to be measured,
ii) identify measurement entity,
iii) identify attributes of the entity that are to be measured,
iv) define new metrics that can be used to measure each attribute.

Table 4: Metrics Definition Process (adapted from Muketha, et al., 2010)

Entity Category	Entity	Attribute	Metric
Determine category of entity that needs to be measured ex. Product	Identify entity to be measured ex. Business process model	Identify attributes of interest ex. Size of Business process model	Define size metric ex. Number of activities (NOA) in the model
Step 1	**Step 2**	**Step 3**	**Step 4**

A business is composed of a set of processes, with each process in turn composed of a set of functional tasks put together to achieve a final goal. Cardoso (2008), states that "high complexity in a process may result in limited understand-ability and more errors, defects, and exceptions, leading processes to need more time to develop, test, and maintain. Therefore, excessive complexity should be avoided."

Whilst the simplification of processes might be the desired ideal, within a business context this is not always possible. Complexity could be dependent on elements external to the process, which reduce the flexibility in simplification within those processes. An example of this could be seen in a product shipping process, whereby approval for dispatch has to go through the finance department as a validation against that customer's credit arrangements.

Parthasarathy, et al. (2006), observed that over the years, metrics have proven effective in controlling complexity, and that proper use of metrics defines and quantifies success or failure, improvement and make useful managerial decisions concerning software or processes. The term complexity has been used by metrics researchers to refer how difficult an entity is to understand. The problem is that very few business process metrics have been defined so far (Ghani, et al., 2008; Cardoso, et al., 2006).

According to Cardoso (2008), the definition of metrics could be done by studying the behaviour of attributes and entities adapted from related fields. Adaptation, which involves technology reuse is a very useful approach because it has the potential to significantly reduce effort of metrics definition.

5.5 Developing the SaaS Complexity Metric (SCM)

Within the complexity research section of this thesis in Section 5.3, it was identified that whilst the complexity being defined is for a business and its processes, the definition being implied is that of the underlying complexity resultant from systems supporting these business processes. For the purpose of developing the SaaS metric, the notions of similarities between software systems and business processes (Ghani, et al., 2008; Vanderfeesten, et al., 2007; Cardoso, et al., 2006) were found to still apply. The author therefore infers that the more complex the business, the more complex the ERP systems required to support it (Spiteri, 2012a).

For the scope of this research, processes that were external to a system were excluded from the framework. Externalities were however still considered throughout the research, particularly from a device integration and system interface perspective. The findings suggest that processes already supported by IT systems should make for an easier transition to a cloud based service.

The complexity metric developed by the author and described in this section, referenced as SCM, aims to synthesise a complexity estimate measure for a business. ***This metric is derived as the number of specific processes nodes and functional elements that an organisation employs***, as described in Spiteri (2012b).

The author then infers a singular relation between how complex the organisation is and the number of interrelated process elements that contribute its business. This definition is extended to suggest that ***the resulting complexity of underlying supporting systems is therefore directly correlated to the complexity of the business itself.***

An example of such a process model and sub-process complexity is represented in Figure 21. Mathematical formulae representing this metric are detailed in the next chapters.

Several researchers have previously put forward metrics directed at measuring complexity for business processes. These metrics take differing approaches in their measures and have a software engineering focus which meant they could not be directly applied to the SCF framework. These limitations are highlighted in the comparison done by Latva-Koivisto (2001) who carries out a review within a business context. The author identified common elements within the context of these metrics, and these have been listed here since applicable to the present work.

Cardoso metrics: Cardoso, et al. (2006, 2008) propose several metrics as follows:

- The Control-Flow Complexity (CFC) metric for measuring control-flow complexity of business process models.
- The Interface Complexity (IC). Where the IC metric is computed as measure of incoming dataflows and output is the outgoing dataflows.
- The Halstead-based Process Complexity (HPC) based on Halstead metrics. The Halstead metrics model is computed from operands and operators present in a program. Specifically, the process Difficulty (D) of this metric is calculated as $D = \frac{n1}{2} \times \frac{N2}{n2}$, where:
    - n1 is the number of distinct operators
    - n2 is the number of distinct operands
    - N2 is the total number of operands

Gruhn and Laue metrics: Gruhn, et al. (2006) have similarly proposed a number of metrics, namely:

- The cognitive weight for business process models. This metric is an adaptation of the Cognitive Functional Size (CFS) proposed by Shao, et al. (2003), but focuses only on control flows. The proponents of cognitive complexity metrics contend that there are three factors that lead to complexity, namely, internal architecture of software, input data flowing into the module and output data flowing out of the module.

- The information flow metric, adapted from Henry, et al., (1981) for business processes. Similar to the IC metric proposed by Cardoso, et al. (2006), but does not include length of the process in its formula.

Vanderfeesten metric: Vanderfeesten, et al. (2007) have proposed a metric called Cross-Connectivity (CC) metric based on cognitive complexity. This is an error prediction metric that measures the strength of the links between process model elements. It is based on the hypothesis that process models are easier to understand and contain fewer errors if they have a high cross-connectivity. In addition to predicting errors, it can measure understandability of a business process model.

Mendling and Neumann metrics: Mendling, et al. (2007) have proposed six error metrics that are closely related to complexity. These metrics are based on graph theory and include size, separability, sequentiality, structure, cyclicity and parallelism.

5.6 Validating the SaaS Complexity Metric

Metrics researchers provide a consensus view that only three steps are needed to define and validate metrics (Serrano, et al., 2002; Soni, et al., 2009). These steps include defining new metrics, validating them theoretically and then validating them empirically.

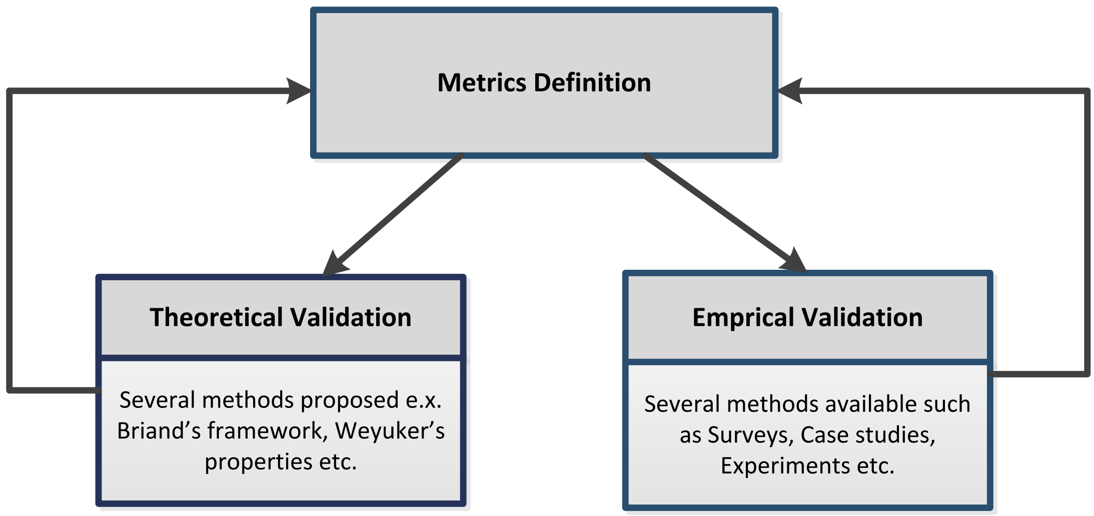


Figure 25: Metrics Definition and Validation Step (adapted from Muketha, 2010)

5.6.1 Theoretical Validation

The main aim of theoretical validation is to establish whether a new metric is structurally sound and follows measurement theory. Whilst there are several theoretical validation methods frequently cited in business process literature, here the author applies two main methods, which include:

1) identifying scale types of new metrics.
2) checking whether the metrics satisfied Briand's generic measurement framework.

Scale types: A knowledge of the scale type of a metric helps determine what transformations are admissible on the metric. It also sheds light on the correlation of the measures that the metric will generate (Fenton, et al., 1997). Scale type fall under four main categories, Nominal, Ordinal, Interval and Ratio. Within the current research, the author applies an interval scale. The dictionary defines this as a scale of measurement of data to which the differences between values can be quantified in absolute but not relative terms, and for which any zero is merely arbitrary.

Briand's generic measurement framework: Briand, et al. (1996) proposed a framework that categorizes metrics into size, length, complexity, coupling and cohesion. Since this SCM metric relates to complexity, the validation applied follows the following process complexity criteria.

The complexity of a process P is a function complexity (P) that is characterised by the following five properties which need to be satisfied by complexity metrics:

- **Complexity 1:** The complexity of a process cannot be negative, but can be null if a system has got no elements.
- **Complexity 2:** The complexity of a process is null if the process has got no structured activities in it.
- **Complexity 3:** The complexity of a process does not depend on the convention chosen to represent the relationships between its elements.
- **Complexity 4:** The complexity of a process is no less than the sum of the complexities of any two of its modules with no relationships in common.
- **Complexity 5:** The complexity of a process composed of two disjoint modules is equal to the sum of the complexities of the two modules.

5.6.2 Empirical Validation

Empirical validation is the empirical investigation of elements of newly designed models before they are transferred to wider practice. As presented by Wieringa (2006), empirical validation research applies to anything designed for a useful purpose, such as a new notation, technique, method, algorithm, device or organisation structure, when such artefact is transferred to practice for use by persons other than the designers using it for their own purposes.

Wieringa (2006) provides some examples where this could apply. For example, a software algorithm that has been transferred to practice when it has been implemented for use within commercially available software. This process follows from initial design, where after some time of redesigned by researchers, engineers and professionals, it may be transferred to practice.

Within this research, the empirical validation of the SCF framework has been evaluated by means of case studies, with the validation taking place through modelling and simulation. It is further suggested by Wieringa (2006), that empirical validation research proceeds by scaling-up from lab conditions to practical real-world conditions. Scaling is carried out in two ways:

1) Addition of realistic conditions of practice.
2) Scaling up to larger sets of subjects.

In line with these metric validation frameworks, the business case studies were collated for validation of the SCF framework and the SCM metric described herein. This conforms to these two scaling criteria, in that actual business data was used in each case; with these being representative of complex businesses in practice.

5.7 Summary

This chapter described the differences in complexity concepts, and detailed how an alternative definition for business complexity was derived, together with the development of the SCM metric to quantify and measure this complexity in a scientific manner. Existing metric validation models were also analysed, and the selection process for the most appropriate model for validating this SCM metric was explained. The validation requirements of the metric, following this model, were then encapsulated within the research methods detailed in the next chapters.

CHAPTER 6

THE SAAS COMPLEXITY FRAMEWORK UNIFIED MODEL

6.1 Introduction

This chapter explains the development of the SaaS Complexity Framework Unified Model (SCFUM), which together with the SCM metric detailed in Chapter 5, constitute the key components of the SCF framework. Step 4 in the research methodology, the SCFUM base model is here derived and described, following the analysis of proprietary ERP systems. The latter part of the chapter describes the application of the SCM metric to the SCFUM, enabling the calculation of a base SaaS Complexity Index (SCI) for this model.

6.2 Analysis of Proprietary ERP Systems

The modelling of complex business systems within hosted internet services informed the requirement for development of a framework to support this industry transition. The SCF framework is intended for both supporting an organisation in evaluating its SaaS implementation approach, and to enable a SaaS service provider to evaluate an industry before investing in that market for the provision of a bespoke SaaS business solution.

As detailed in previous chapters, whilst the complexity being defined here is for a business and its processes, the definition implied is that of complexity resultant from system supported business processes. That is, complexity for processes inherent within the business that are supported by one or more IT systems. For these reasons, the case study analysis finds that processes already supported by IT systems would make for an easier transition to a cloud based service, over an organisation making the transition from the ground-up.

An ERP system is here being considered as an example of a complex business system. The term ERP traditionally referred to enterprise wide software for larger organisations. Its modular features allow planning in the use of organisational wide resources, and these systems are thus mostly used in larger, more industrial, types of businesses. The use of ERP has however changed (Tech-Faq, 2009; Web Based ERP, 2006), and has over the years been applied to a very comprehensive range of business types, irrespective of the size of industry or organisation, as highlighted in Table 5. This has been facilitated by the modular nature and innovation of ERP vendors and systems which enable selective deployment levels, matching complexity of the software itself to that of the organisation. ERP systems can cover a wide range of functions often deployed as modules.

Many proprietary ERP systems adopt a similar architecture. Originally, these business functions were managed through their own disparate applications, where each of these departments would typically have their own specialist software systems optimised for each specific role, however these can now be unified and integrated within the ERP system. An ERP therefore is a mechanism to integrate enterprise data and processes of an organisation into one single system, a software base that serves the needs of users in finance as it does for those in human resources or in the warehouse (Welch and Kordysh, 2007).

A second-tier distribution mechanism has become established where ERP applications have been extended to specialise in specific market sectors, for example servicing the fresh food processing industry.

ERP systems will usually have many components including hardware and software to achieve this integration (Hvolby and Trienekens, 2010; Tech-Faq, 2009). Most systems use a unified database to store data for these various functions throughout the organisation, such that the various departments can more easily share and communicate information. This integrated approach can have a tremendous benefit with respect to meeting market requirements and dealing with the challenges for companies in the global knowledge economy (Wailgum, 2010).

Table 5: ERP Adoption by Industry Type (adapted from Shin, 2006)

ERP solution adoption by Industry types
Agriculture, forestry, fishing and mining
Food, tobacco, textile and apparel
Lumber, wood, paper and printing
Chemicals and petroleum
Primary and fabricated metal, instruments
Electronics and other electronic equipment
Motor Vehicle, Transportation Equipment and other manufacturing
Electric, gas, Sanitary services and construction
Wholesale and Retail trade
Hotels, other lodging and Restaurants
Transportation and communication
Financial institution, Insurance and business services
Public and personal services

Another key element of the SCF framework was the development of a unified model, referenced as the SaaS Complexity Framework Unified Model (SCFUM). The SCFUM model is geared to represent a unified view of various business processes and functions, common across industries. The basis of this unified business model was derived through the review of the common elements of off-the-shelf ERP systems, together with the case study reviews. Through this process, two distinct ERP systems were primarily referenced as representative of Tier-1 and Tier-2 solutions, these being the Microsoft Dynamics Navision and AX systems. Whilst there is a wide selection of ERP systems on the market, the ones that could support large complex businesses are limited to a few key ones. The two systems described in Section 4.4 provided a strong representation of the wider ERP industry solutions at this level, and therefore the common elements shared across these.

6.3 Developing the SCF Unified Model (SCFUM)

The methodology applied for the development of a system-agnostic unified model within the SCF framework, required an initial review and analysis of established ERP systems described in Section 4.4. This offered a spectrum of complexity in business processes which have been extended to a wide range of industries (Alpern, 2010). ERP comprises a multitude of business functions within one system, and additionally provides the facility of interfacing with other systems within the business (Botta-Genoulaz, 2005; Shin, 2006). These generic solutions could therefore be extended to support scalable global business.

As defined earlier in the thesis, an ERP implementation might not necessarily imply a complex business system. Partial modules can be implemented separately to cover specific aspects of the business (Mabert, Et al. 2001; Chang, 2010). The author therefore determined that a better understanding of business complexity is key to ensure a better translation of business requirements and support within a SaaS solution.

The alternative complexity definition applied here is that a complex business process, within an ERP context, is one that requires the use of most of that system's functional elements. To this effect, the research analysis required garnering an in-depth understanding of the key business processes and functionalities that were representative of ERP systems. The methodology then applied the following approach in defining the SCFUM model:

- Extraction of the functional elements comprising each ERP system.
- Investigation of the functional elements and interaction within the wider processes.
- Compilation of common functional elements and related processes across systems in a unified model.

These business elements were modelled as follows:

1) The **Process Functional Elements** – these being the lowest level of detail representing the process functional elements of the business data entities.
2) The **Process Nodes** – these being the general processes made up as a collation of process functional elements, forming a business module or functional group.

The resultant unified model formed the basis of the first iteration of the SCF framework. An example of the unified business processes nodes, and functional elements is represented in Table 6. The final version of the SCFUM model can be found in Appendix A – SCFUMv2, though this could be expanded further for future applications.

Table 6: SCFUM v1: Business Process Functional Elements (Spiteri, 2012b)

Process Node	**Business Functional Element**	**Description**
Finance	Budgets	Financial budgeting for expected value of revenue and expenses
	General Journal	Processing of financial transactions to the general ledger
	Chart of Accounts	Financial management reporting
	Intercompany Outbox	Support for intercompany transactions in a multi-company environment
	Intercompany Inbox	Support for intercompany transactions in a multi-company environment
	Payment Journals	Processing of payments to suppliers
	Deposits	Processing of bank deposits
	Bank Account Reconciliations	Processing of financial bank reconciliations
	Cash Receipt Journals	Processing of cash receipts into the business
	Bank Accounts	Management of business bank accounts
	Fixed Assets	Management of business fixed assets records
	Insurance	Management of insurances in place for Fixed assets
	FA Journals	Processing of fixed asset related transactions
	Insurance Journals	Processing of insurance related transactions
	FA Registers	Reporting on audit trail for Fixed asset transactions
	General Ledger Registers	Audit trail for financial transactions
HRM	Human Resources	Management of the organisation's human resources for internal and statutory reasons
	Causes of absence	Process for recording resource absence and reasons
	Causes of inactivity	Process for recording resource inactivity and reasons
	Relatives	Management of employee records for next of kin
	Unions	Management of employee records for union affiliations
	Employment Contracts	Management of employee contracts of employment

	Qualifications	Management of employee qualification records
Procurement	Requisitions	Purchase requisition process for supply replenishment
	Purchase Orders	Purchase processing for supply replenishment
	Purchase Invoices	Processing of supplier invoices for product or services receipt
	Purchase Quotes	Processing and record of quotes received for required product or service
	Purchase Order Archive	Archiving of various versions of purchase documents
	Vendors	Management of supplier records
	Purchase Prices	Management of the business purchase prices for various suppliers
Manufacturing	Production Forecasts	Volume forecasting for meeting of sales demand through production or product manufacturing
	Production BOM	Management of bill of materials for production
	Production Routings	Management of product routings for production
	Machine Centres	Management of machine centres making up production routings
	Work Centres	Management of work centres making up production routings
	Capacity Journals	Processing of capacity transactions for Machine and work centres against production
	Planning Worksheets	Process for managing supply and demand, and action results within business
	Planned Production Orders	Process for medium term planning production requirements for the shop floor
	Firm Planned Production Orders	Process for shorter term planning production requirements for the shop floor
	Released Production Orders	Process for the release of production orders for the shop floor to action
	Production Schedule	Management of the schedule for the organisation's production requirements
	Consumption	Process for recording consumption of product within production process
	Output	Process for recording output of product within the production process
	Finished Production Orders	Process for closing off a completed production order
	Work Shifts	Management of various work shifts a business might have particularly in production
	Shop Calendars	Management of shop floor work calendars
	Resources	Management of resources to the business particularly for production

Sales	Sales Orders	Sales process for product or service for generation of revenue
	Blanket Sales Orders	Sales agreements with a customer in place over a period
	Sales Invoices	Processing of customer invoices for product or services delivered
	Sales Quotes	Processing and submission of customer quotes in support of sale of product or service
	Sales Order Archive	Archiving of various versions of sales documents
	Salespeople	Management of Sales persons' records
	Contacts	Management of business contacts
	Sales Forecasts	Management of sales forecast data within the business
	Sales Prices	Management of the business product sales prices, discounts etc.
	Customers	Management of customer records
Sales / Procurement	Return Orders	Product return process back to Supplier
	Credit Notes	Processing of credit adjustments to customer or supplier accounts
	Payment Terms	Management of payment terms
	Payment Methods	Management of payment methods for purchases and sales
Warehouse	Transfer Orders	Transfer of stock from one storage location to another within the business
	Item Journals	Processing of stock counts, stock adjustments and stock movements
	Inventory	Audit trail for Inventory transactions
	Warehouse Locations	Management of the organisation's product storage locations
	Warehouse Receipts	Processing of product receipts within storage locations
	Warehouse Shipments	Processing of product dispatch from storage locations
	Warehouse Put-aways	Processing of directed product put-away within a storage location
	Warehouse Picks	Processing of directed product picks from a storage location
	Warehouse Movements	Processing of movement of goods within storage locations
	Bin Contents	Management of bins and bin content within a storage location

6.3.1 Terminology Variances across Organisations and Systems

A consideration of this work was that different organisations and their supporting systems are likely to have developed their own terminology for describing similar business processes and functions. These differences in terms and labels are similarly evident when comparing processes across industries. The author therefore normalised these ontological differences for a common evaluation of complexity within the SCFUM model. Some such examples being Stock and Inventory, or Production and Manufacturing.

There is generally an underlying layer of commonality in businesses operations and therefore processes and functions could still be comparable. This commonality enabled the development and proliferation of generic ERP systems as described previously. In this respect, the framework process and function list as defined in Table 6, applies terminology from the proprietary reference ERP systems, the business case studies, and interspersed with the author’s own terminology, based on professional experience.

This approach would allow the targeted application of the SCF framework to a range of businesses and business systems, through the application of a semantic mapping exercise. The reverse-mapping of the framework then validated and normalised these business functions and processes.

6.4 Summary

This chapter described the development of the SCFUM model, and how it has been derived through analysis of proprietary ERP systems, referenced as exemplars of complex business systems. Highlighting the variances in terminology for common elements across organisations and business systems, the SCFUM model applied a generic ontological approach to the terms used for its structure of Process Nodes and Process Functional Elements. The aim for this was to ensure the possibility of cross-industry application of the framework.

CHAPTER 7

APPLICATION AND VALIDATION OF THE FRAMEWORK

7.1 Introduction

The objective of this chapter is to explain the application of the SCF framework to business case studies as proof of concept, enabling its validation through the resultant output. This being Step 5 of the research methodology, the methods applied for the execution of this process are expanded upon. This then leads to methodology Step 6, where the author describes how analysis and evaluation of the application results allowed for the model to be calibrated for more comprehensive business coverage.

The business case selection process is described, together with an overview of the data file structures created in the process of compiling and analysing these case studies.

Extracts from the case study datasets were used to detail the SCFUM model mapping process and the SCM metric calculation. This application and validation of the SCF framework were carried out through a step by step worked example. The baseline SCI was calculated for the SCFUM model, explaining the mathematics used in deriving these results, and how these were applied in the interpretation of the datasets. The full business datasets related to this work are provided in digital format as an addendum to the thesis.

7.2 Methodology for Validating the Framework

To test and validate the SCF framework, case studies using actual business datasets were used. These business case datasets were comprised of anonymised extracts of actual organisational Reference and Transactional data. These were each derived from nine complex organisations, operating within the food processing industry.

As represented in Figure 26, the method applied in this step had the following two key objectives:

1) Validate the SCF framework for real-life commercial applications, and
2) Revise SCFUM model for any shortcomings based on analysis of the data.

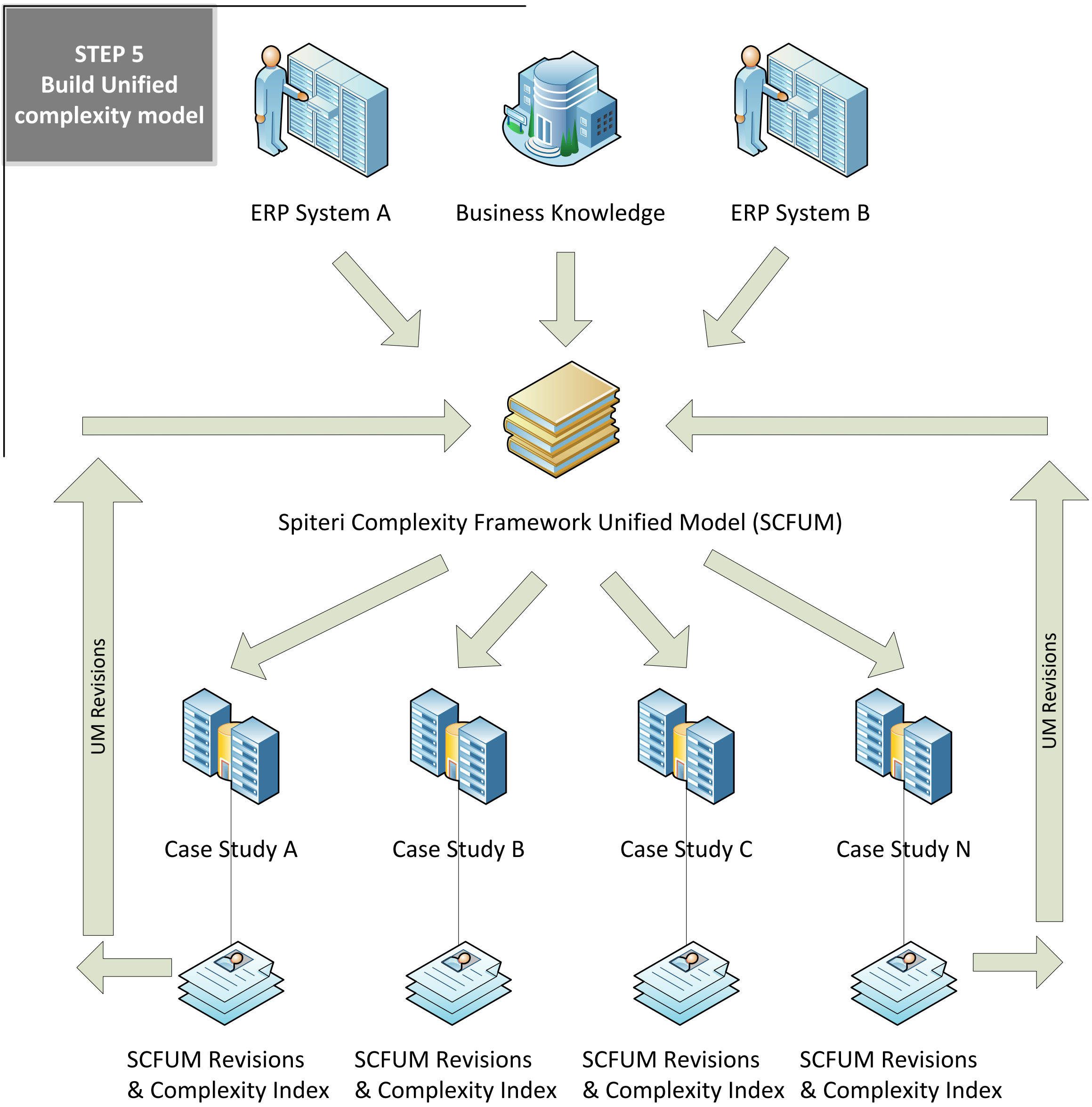


Figure 26: Application of the Unified Model (Spiteri, 2014)

The resultant dataset from this analysis enabled a review of overall business case complexity, and a comparison of these to the baseline complexity index. The method for deriving the complexity index for each case required the individual process elements within these to be mapped to the SCFUM model. This process was done for all process elements iteratively, covering each specific case dataset.

This process was done iteratively such that the analyses results could be used to revise the model to cover for any resultant variances identified within each case. These iterative steps are further described later in this chapter. The result was a more robust and comprehensive model. The main methods within this part of the methodology are represented in Figure 27.

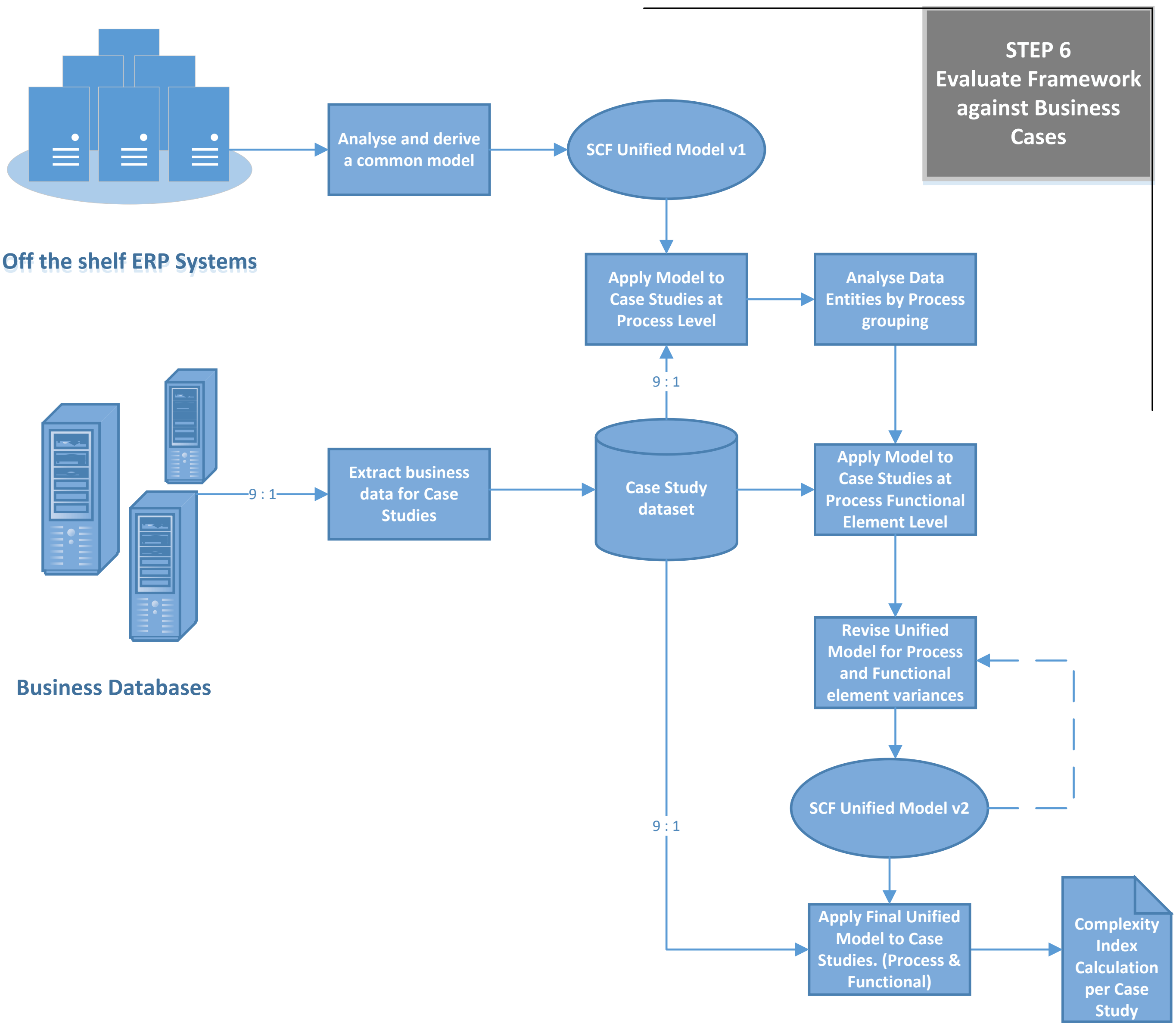


Figure 27: SCF Unified Model Validation Methodology (Spiteri, 2014)

Within this methodology, a three level, iterative mapping process was therefore incorporated, represented in Figure 27, these being:

- **Process Functional Elements** as the lowest level of functional detail to be mapped.
- **Process Nodes** as process groups made up by the various process functional elements.
- **SCF Unified Model** representing the SCF framework's common set of process and functional elements, unified across systems and industries.

7.3 Data File Structures in the Framework Application

Several data files were generated in the application of the SCF framework and analyses of the business case study data. Microsoft Excel was the primary tool used for this purpose, both for manipulating the data, and for analysing results through the powerful business intelligence functionality that comes with it.

Figure 28 details the main file structures created within the course of this research and the sequence these were created in. Appendix C – Case Data Folder Structure, outlines the folder structure and contents of the complete business case datasets, provided in digital format as an addendum to this thesis. The research data file structure is as follows:

(1) **Case#_rawdata.xls** – the raw data for each business case was stored in nine such files, with the # symbol within the file name representing the Case reference. This symbolic representation is used throughout the Figure 28 diagram file labelling. An extract from one such file can be seen in Table 8.

(2) **Case#_mergedata.xls** – the raw data in many of the cases comprised multiple companies within the same enterprise. This was a further representation of the complexity of the selected businesses. To facilitate the application of the framework to each enterprise business case, each raw dataset was merged within nine such case files. For each case reference, these files grouped all the companies' data as one per business case.

(3) **Case#_normdata.xls** – the case datasets were business snapshots with varying time periods. To enable a better comparative analysis of the datasets, the data for each case reference was normalised to represent a month of business transacting. Thus, each business case had such a file generated to hold this normalised data.

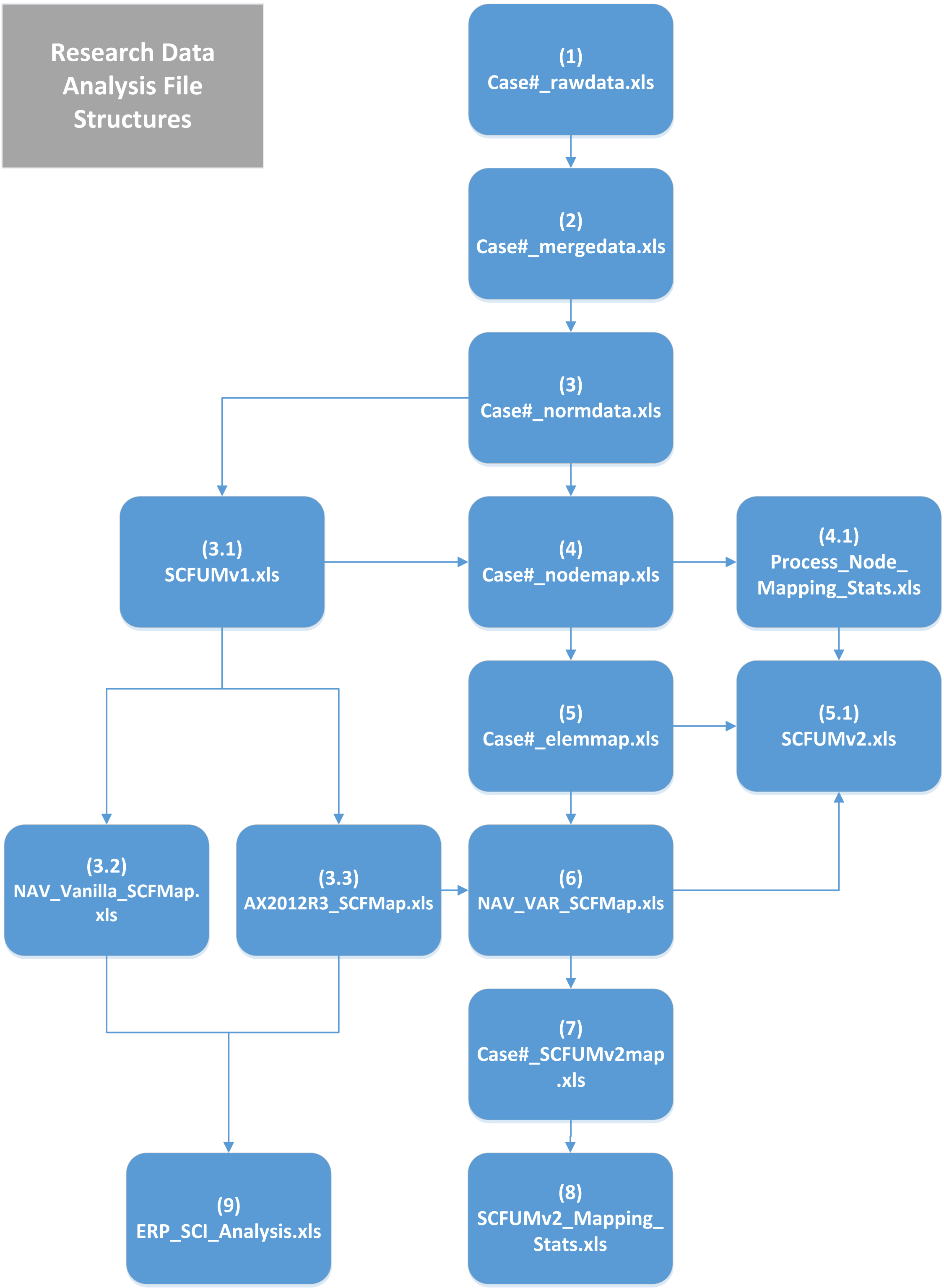


Figure 28: Research Data Analyses File Structures

(3.1) **SCFUMv1.xls** – the SCFUM model was developed and managed within this file. The first iteration of the framework was developed in parallel to the normalisation of the business cases. This was introduced at this point within the sequence since it serves as an input for the next parts of the case analysis process.

(3.2) **NAV_Vanilla_SCFMap.xls** – this file was created in parallel to the main case analysis strand. Here the SCFUM was applied to Microsoft NAV vanilla system, with the results saved within this file. Note, within this context, the term "Vanilla" is used to represent a non-customised solution.

(3.3) **AX2012R3_SCFMap.xls** – the SCFUM was likewise applied to Microsoft AX vanilla system, with the results saved within this file. Both AX and NAV systems were used as exemplars of ERP systems in general, and the results informed further comparative analyses later in the process.

(4) **Case#_nodemap.xls** – there is one such file per business case. This contains results of the mapping and map management of the SCF Process Node to that business dataset.

(4.1) **Process_Node_Mapping_Stats.xls** – this file holds the statistics for the case results following the Node mapping exercise. The case results were here collated together and analysed, enabling variances encountered to be identified.

(5) **Case#_elemmap.xls** – there is one such file per business case. This contains the results from the mapping and map management of the low-level SCF Process Elements, as applied to that business dataset.

(5.1) **SCFUMv2.xls** – the calibrated SCFUM model was iteratively revised and managed within this file. This file holds the final version of the system-agnostic SCFUM model derived from this research.

(6) **NAV_VAR_SCFMap.xls** – this file holds the detailed analysis of the SCFUM model variances in Process Nodes and Elements between the business cases and the vanilla NAV ERP system. This file additionally identifies the industry specific, and business specific process nodes and elements.

(7) **Case#_SCFUMv2map.xls** – there was one such file per business case, identified through the case reference. Each of these files contains the results from the

SCFUMv2 mapping and map management, for Process Nodes and Elements, as applied to that business dataset.

(8) **SCFUMv2_Mapping_Stats.xls** – this analysis file collates the results derived from the “Case#_SCFUMv2map.xls” process, and through the generation of pivot reports derives various stats from this output, to highlight the capabilities of the framework. Various detailed representations of derived complexity per business case are managed here, as well as quantifying additional strategic commercial elements applicable in modelling a complex business on to the Cloud. These results validated the potential use of the SCF framework within a real-life business environment.

(9) **ERP_SCI_Analysis.xls** – this analysis file details a comparative investigation of the vanilla NAV and AX ERP systems, derived through the generation of pivot reports using process results from “NAV_Vanilla_SCFMap.xls” and “AX2012R3_SCFMap.xls”. The output of this analysis served to highlight that the use of the framework is not limited to cloud modelling scenarios, but has the possibility of wider commercial applications for the business.

7.4 Case Study Empirical Validation Example

To enable the calculation of representative results in validating the SCF framework and the related complexity metric, the business case studies had to be examples of complex organisations.

7.4.1 Case Study Business Selection and Raw Datasets

The selection of the individual businesses was done based on the application of the qualitative complexity criteria, defined in Section 5.3. These criteria being the qualitative process and functional coverage evident within each data-set. The aim was to ensure the selection of business data with the appropriate level of complexity. Applying this defining criterion that a complex business is one that employs a higher number of interrelated processes, and underpinned by the author’s professional experience, nine such organisations were identified. These all displayed a wide range of interrelated processes, represented within their underlying IT systems, and therefore each demonstrating complexity in their own stead.

Table 7: Business Case Study Dataset Details

Case Ref	No. of Entities	No. of Companies	System Version	Industry	Total No. of Records	Avg. Record Size (B)	Total Dataset Size (KB)	Entities with records	Merged No. of Entities	Merged Entities w/records
Case A	4098	3	Navision 6 SP1	Fruit Packing and distribution	7,654,844	7,017	1,274,016	1,676	1,388	876
Case B	5980	4	Navision 6 R2 NA	Fruit & Veg Packing	81,773,849	7,051	26,801,888	2,119	1,525	635
Case C	12726	7	Navision 6 R2 NA	Fruit & Veg Packing	194,371,349	7,968	60,098,456	2,685	1,860	623
Case D	2946	2	Navision 6 SP1 NA	Shell Egg processing	115,047,556	8,518	24,968,704	814	1,504	544
Case E	9645	7	Navision 6 R2	Fruit processing	10,028,165	7,909	1,580,008	3,143	1,407	566
Case F	7230	5	Navision 6 R2	Poultry Processing	748,290	9,148	553,968	2,787	1,482	865
Case G	1607	1	Navision 5	Veg Packing	171,192,346	6,550	54,661,976	543	1,607	543
Case H	3153	2	Navision 5	Veg Packing	51,261,279	7,735	14,963,048	497	1,600	471
Case I	1509	1	Navision 5 SP1	Food processing	50,157,879	7,794	12,088,184	371	1,509	371

In each case, a data extract was taken as a snapshot from each of the business's respective back-end databases. Each dataset extract was for database entity level metadata covering Reference and Transactional data over several months of commercial operation. The individual business cases were then anonymised, and given a case reference indicator, this being a letter from A to I. This data was stored in the **Case#_rawdata.xls** filesets, where the # sign represents a case reference letter.

An extract from one such raw dataset can be found in Table 8, to provide a view on the kind of data retrieved. Each column within this table represents the following data elements:

> **Case Reference** – This would be defining the business entity within the database. For the purposes of this research the dataset has been anonymised with a Case reference applied to the data throughout the analyses, and ensuring there is no company identifying references within any data referenced thereafter.
>
> **Entity No.** – This is the table reference for the entity within the database, represented as a unique number.
>
> **Entity Name** – This represents the name of the entity within the database structure.
>
> **No. of records** – This represents the number of records for that entity within the dataset.
>
> **Record Size** – This represents an indicative average number of bytes per record in the entity.
>
> **Entity Size (KB)** – This represents total size of the dataset entity, in kilobytes, reported by the database.

Taking the "No. of records" value together with the "Record size" value returns an indication of the persistent storage space required. The databases metadata reported on these values independently however (not as a derivative calculation across fields). As a result, slight differences were therefore noted between calculated and actual size (KB) values. This dataset extract from CASE G, is used throughout the step-by step worked example over the next sections.

Table 8: CaseG_rawdata.xls Extract

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	5603	FA Setup	1	16,384	16
Case G	5604	FA Posting Type Setup	0		0
Case G	5605	FA Journal Setup	0		0
Case G	5606	FA Posting Group	0		0
Case G	5607	FA Class	0		0
Case G	5608	FA Subclass	0		0
Case G	5609	FA Location	0		0
Case G	5611	Depreciation Book	0		0
Case G	5612	FA Depreciation Book	0		0
Case G	5615	FA Allocation	0		0
Case G	5616	Maintenance Registration	0		0
Case G	5617	FA Register	0		0
Case G	5619	FA Journal Template	2	8,192	16
Case G	5620	FA Journal Batch	2	8,192	16
Case G	5621	FA Journal Line	0		0
Case G	5622	FA Reclass. Journal Template	1	16,384	16
Case G	5623	FA Reclass. Journal Batch	1	16,384	16
Case G	5624	FA Reclass. Journal Line	0		0
Case G	5625	Maintenance Ledger Entry	0		0
Case G	5626	Maintenance	0		0
Case G	5628	Insurance	0		0
Case G	5629	Ins. Coverage Ledger Entry	0		0

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	5630	Insurance Type	0		0
Case G	5633	Insurance Journal Template	0		0
Case G	5634	Insurance Journal Batch	0		0
Case G	5635	Insurance Journal Line	0		0
Case G	5636	Insurance Register	0		0
Case G	5637	FA G/L Posting Buffer	0		0
Case G	5640	Main Asset Component	0		0
Case G	5641	FA Buffer Projection	0		0
Case G	5642	Depreciation Table Header	0		0
Case G	5643	Depreciation Table Line	0		0
Case G	5644	FA Posting Type	0		0
Case G	5645	FA Date Type	0		0
Case G	5646	Depreciation Table Buffer	0		0
Case G	5647	FA Matrix Posting Type	0		0
Case G	5648	FA Allocation Dimension	0		0
Case G	5649	FA Posting Group Buffer	0		0
Case G	5650	Total Value Insured	0		0
Case G	5700	Stockkeeping Unit	27	2,427	64
Case G	5701	Stockkeeping Unit Comment Line	0		0
Case G	5714	Responsibility Center	0		0
Case G	5715	Item Substitution	0		0
Case G	5716	Substitution Condition	0		0
Case G	5717	Item Cross Reference	266	1,509	392

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	5718	Nonstock Item	1	65,536	64
Case G	5719	Nonstock Item Setup	1	16,384	16
Case G	5720	Manufacturer	0		0
Case G	5721	Purchasing	0		0
Case G	5722	Item Category	15	1,092	16
Case G	5723	Product Group	9	1,820	16
Case G	5740	Transfer Header	0		24
Case G	5741	Transfer Line	0		136
Case G	5742	Transfer Route	0		0
Case G	5744	Transfer Shipment Header	2595	218	552
Case G	5745	Transfer Shipment Line	2596	426	1080
Case G	5746	Transfer Receipt Header	2585	193	488
Case G	5747	Transfer Receipt Line	2586	472	1192
Case G	5748	Inventory Comment Line	1	16,384	16
Case G	5765	Warehouse Request	14554	205	2912
Case G	5766	Warehouse Activity Header	3	16,384	48
Case G	5767	Warehouse Activity Line	0		328
Case G	5768	Whse. Cross-Dock Opportunity	88	1,303	112
Case G	5769	Warehouse Setup	1	16,384	16
Case G	5770	Warehouse Comment Line	0		16
Case G	5771	Warehouse Source Filter	0		0
Case G	5772	Registered Whse. Activity Hdr.	25705	218	5472
Case G	5773	Registered Whse. Activity Line	904191	483	426768

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	5790	Shipping Agent Services	19	862	16
Case G	5800	Item Charge	2	24,576	48
Case G	5802	Value Entry	10402133	1,230	12493320
Case G	5803	Item Journal Buffer	0		0
Case G	5804	Avg. Cost Adjmt. Entry Point	247444	100	24104
Case G	5805	Item Charge Assignment (Purch)	0		0
Case G	5809	Item Charge Assignment (Sales)	0		48
Case G	5810	Rounding Residual Buffer	0		0
Case G	5811	Post Value Entry to G/L	0		280
Case G	5813	Inventory Posting Setup	117	140	16
Case G	5814	Inventory Period	46	356	16
Case G	5815	Inventory Period Entry	35	468	16
Case G	5820	Cost Element Buffer	0		0
Case G	5821	Item Statistics Buffer	0		0
Case G	5822	Invt. Post to G/L Test Buffer	0		0
Case G	5823	G/L - Item Ledger Relation	21916774	56	1208048
Case G	5832	Capacity Ledger Entry	909283	454	403560
Case G	5840	Standard Cost Worksheet Name	1	16,384	16
Case G	5841	Standard Cost Worksheet	0		0
Case G	5845	Inventory Report Header	0		0
Case G	5846	Inventory Report Entry	0		0
Case G	5847	Average Cost Calc. Overview	0		0
Case G	5848	Cost Share Buffer	0		0

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	5900	Service Header	0		0
Case G	5901	Service Item Line	0		0
Case G	5902	Service Line	0		0
Case G	5903	Service Order Type	0		0
Case G	5904	Service Item Group	0		0
Case G	5905	Service Cost	0		0
Case G	5906	Service Comment Line	0		0
Case G	5907	Service Ledger Entry	0		0
Case G	5908	Warranty Ledger Entry	0		0
Case G	5909	Service Shipment Buffer	0		0
Case G	5910	Service Hour	0		0
Case G	5911	Service Mgt. Setup	1	16,384	16
Case G	5912	Service Document Log	0		0
Case G	5913	Loaner	0		0
Case G	5914	Loaner Entry	0		0
Case G	5915	Fault Area	0		0
Case G	5916	Symptom Code	0		0
Case G	5917	Fault Reason Code	0		0
Case G	5918	Fault Code	0		0
Case G	5919	Resolution Code	0		0
Case G	5920	Fault/Resol. Codes Rlship.	0		0
Case G	5927	Repair Status	0		0
Case G	5928	Service Status Priority Setup	0		0

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	5929	Service Shelf	0		0
Case G	5933	Service Order Posting Buffer	0		0
Case G	5934	Service Register	0		0
Case G	5935	Service E-Mail Queue	0		0
Case G	5936	Service Document Register	0		0
Case G	5940	Service Item	0		0
Case G	5941	Service Item Component	0		0
Case G	5942	Service Item Log	0		0
Case G	5943	Troubleshooting Header	0		0
Case G	5944	Troubleshooting Line	0		0
Case G	5945	Troubleshooting Setup	0		0
Case G	5950	Service Order Allocation	0		0
Case G	5952	Resource Location	0		0
Case G	5954	Work-Hour Template	0		0
Case G	5955	Skill Code	0		0
Case G	5956	Resource Skill	0		0
Case G	5957	Service Zone	0		0
Case G	5958	Resource Service Zone	0		0
Case G	5964	Service Contract Line	0		0
Case G	5965	Service Contract Header	0		0
Case G	5966	Contract Group	0		0
Case G	5967	Contract Change Log	0		0
Case G	5968	Service Contract Template	0		0

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	5969	Contract Gain/Loss Entry	0		0
Case G	5970	Filed Service Contract Header	0		0
Case G	5971	Filed Contract Line	0		0
Case G	5972	Contract/Service Discount	0		0
Case G	5973	Service Contract Account Group	0		0
Case G	5989	Service Shipment Item Line	0		0
Case G	5990	Service Shipment Header	0		0
Case G	5991	Service Shipment Line	0		0
Case G	5992	Service Invoice Header	0		0
Case G	5993	Service Invoice Line	0		0
Case G	5994	Service Cr.Memo Header	0		0
Case G	5995	Service Cr.Memo Line	0		0
Case G	5996	Standard Service Code	0		0
Case G	5997	Standard Service Line	0		0
Case G	5998	Standard Service Item Gr. Code	0		0
Case G	6080	Service Price Group	0		0
Case G	6081	Serv. Price Group Setup	0		0
Case G	6082	Service Price Adjustment Group	0		0
Case G	6083	Serv. Price Adjustment Detail	0		0
Case G	6084	Service Line Price Adjmt.	0		0
Case G	6502	Item Tracking Code	3	5,461	16
Case G	6504	Serial No. Information	0		0
Case G	6505	Lot No. Information	285694	454	126648

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	6506	Item Tracking Comment	6110	106	632
Case G	6507	Item Entry Relation	366958	165	58992
Case G	6508	Value Entry Relation	357362	123	42848
Case G	6509	Whse. Item Entry Relation	27446	143	3832
Case G	6520	Item Tracing Buffer	0		496
Case G	6521	Item Tracing History Buffer	0		0
Case G	6529	Record Buffer	0		0
Case G	6550	Whse. Item Tracking Line	234	945	216
Case G	6635	Return Reason	1	16,384	16
Case G	6650	Return Shipment Header	419	978	400
Case G	6651	Return Shipment Line	859	1,001	840
Case G	6660	Return Receipt Header	1016	863	856
Case G	6661	Return Receipt Line	3255	921	2928
Case G	6670	Returns-Related Document	0		0
Case G	6800	Employee Portal Setup	1	16,384	16
Case G	6804	EP Group	0		0
Case G	6805	EP Web Part Request	0		0
Case G	6806	EP WP Request Table Tab	0		0
Case G	6807	EP WP Request Tab Field	0		0
Case G	6809	EP WP Request Table Action	0		0
Case G	6810	EP WPR Table Edit Criterion	0		0
Case G	6811	EP WP Request Table	0		0
Case G	6813	EP WPR Table Action Filter	0		0

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	6815	EP WPR Header/Line Connection	0		0
Case G	6822	EP WP Request Table Sort Key	0		0
Case G	6824	EP WPR Header Create Criterion	0		0
Case G	6825	EP Temporary Filter	0		0
Case G	6827	EP WPR Field Lookup	0		0
Case G	6828	EP WPR Field Lookup Condition	0		0
Case G	6829	EP WPR Field Lookup Mapping	0		0
Case G	6832	EP WPR Table Filter Field	0		0
Case G	6833	EP WPR Table Filter Key	0		0
Case G	6835	EP SharePoint Image Path	0		0
Case G	6836	EP Search Table	0		0
Case G	6837	EP Search Field	0		0
Case G	6838	EP Search Display Field	0		0
Case G	6839	EP Search Configuration	0		0
Case G	6840	EP Group/User	0		0
Case G	6841	EP User Login	0		0
Case G	6842	EP User	0		0
Case G	6843	EP Search Result	0		0
Case G	6850	EP Caption	0		0
Case G	6870	EP Appln. Server Setup	0		0
Case G	6872	EP Trusted Site	0		0
Case G	7002	Sales Price	1195	446	520
Case G	7004	Sales Line Discount	0		0

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	7012	Purchase Price	3	10,923	32
Case G	7014	Purchase Line Discount	0		0
Case G	7023	Sales Price Worksheet	0		0
Case G	7030	Campaign Target Group	0		0
Case G	7110	Analysis Field Value	0		0
Case G	7111	Analysis Report Name	0		0
Case G	7112	Analysis Line Template	3	5,461	16
Case G	7113	Analysis Type	0		0
Case G	7114	Analysis Line	0		0
Case G	7116	Analysis Column Template	3	5,461	16
Case G	7118	Analysis Column	0		0
Case G	7132	Item Budget Name	2	8,192	16
Case G	7134	Item Budget Entry	0		0
Case G	7135	Item Budget Dimension	0		0
Case G	7136	Item Budget Buffer	0		0
Case G	7152	Item Analysis View	3	5,461	16
Case G	7153	Item Analysis View Filter	0		0
Case G	7154	Item Analysis View Entry	8135	227	1800
Case G	7156	Item Analysis View Budg. Entry	0		0
Case G	7158	Analysis Dim. Selection Buffer	0		0
Case G	7159	Analysis Selected Dimension	2	16,384	32
Case G	7190	Sales Shipment Buffer	0		0
Case G	7300	Zone	5	3,277	16

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	7301	Warehouse Employee	449	292	128
Case G	7302	Bin Content	857	1,434	1200
Case G	7303	Bin Type	4	4,096	16
Case G	7304	Warehouse Class	2	8,192	16
Case G	7305	Special Equipment	0		0
Case G	7307	Put-away Template Header	2	8,192	16
Case G	7308	Put-away Template Line	5	3,277	16
Case G	7309	Warehouse Journal Template	3	5,461	16
Case G	7310	Warehouse Journal Batch	3	10,923	32
Case G	7311	Warehouse Journal Line	2	94,208	184
Case G	7312	Warehouse Entry	9674973	578	5464768
Case G	7313	Warehouse Register	4866798	79	375744
Case G	7316	Warehouse Receipt Header	6973	177	1208
Case G	7317	Warehouse Receipt Line	2203	718	1544
Case G	7318	Posted Whse. Receipt Header	26039	227	5776
Case G	7319	Posted Whse. Receipt Line	29659	539	15608
Case G	7320	Warehouse Shipment Header	980	276	264
Case G	7321	Warehouse Shipment Line	644	1,183	744
Case G	7322	Posted Whse. Shipment Header	31668	257	7960
Case G	7323	Posted Whse. Shipment Line	207467	479	96968
Case G	7324	Whse. Put-away Request	36	455	16
Case G	7325	Whse. Pick Request	3577	69	240
Case G	7326	Whse. Worksheet Line	0		104

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	7327	Whse. Worksheet Name	4	4,096	16
Case G	7328	Whse. Worksheet Template	3	5,461	16
Case G	7330	Bin Content Buffer	0		0
Case G	7331	Whse. Internal Put-away Header	0		0
Case G	7332	Whse. Internal Put-away Line	0		0
Case G	7333	Whse. Internal Pick Header	0		0
Case G	7334	Whse. Internal Pick Line	0		0
Case G	7335	Bin Template	0		0
Case G	7336	Bin Creation Wksh. Template	1	16,384	16
Case G	7337	Bin Creation Wksh. Name	1	16,384	16
Case G	7338	Bin Creation Worksheet Line	0		32
Case G	7340	Posted Invt. Put-away Header	0		0
Case G	7341	Posted Invt. Put-away Line	0		0
Case G	7342	Posted Invt. Pick Header	0		0
Case G	7343	Posted Invt. Pick Line	0		0
Case G	7354	Bin	113	870	96
Case G	7380	Phys. Invt. Item Selection	0		0
Case G	7381	Phys. Invt. Counting Period	0		0
Case G	7600	Base Calendar	0		0
Case G	7601	Base Calendar Change	0		0
Case G	7602	Customized Calendar Change	0		0
Case G	7603	Customized Calendar Entry	0		0
Case G	7604	Where Used Base Calendar	0		0

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	7700	Miniform Header	0		0
Case G	7701	Miniform Line	0		0
Case G	7702	Miniform Function Group	0		0
Case G	7703	Miniform Function	0		0
Case G	7704	Item Identifier	0		0
Case G	7709	XMLQueue	0		0
Case G	8000	Notification Setup	1	16,384	16
Case G	8001	Notification	0		0
Case G	8002	Notification Worksheet Batch	0		0
Case G	8003	Notification Worksheet Line	0		0
Case G	8004	Notification Line	0		0
Case G	8005	Notification Log Entry	0		0
Case G	8610	Setup Questionnaire	0		0
Case G	8611	Question Area	0		0
Case G	8612	Question	0		0
Case G	8613	Migration Table	1	16,384	16
Case G	8614	Migration Record	0		24
Case G	8615	Migration Data	0		72
Case G	8616	Migration Table Field	153	107	16
Case G	8617	Migration Data Error	0		16
Case G	8618	Data Template Header	0		0
Case G	8619	Data Template Line	0		0
Case G	8620	Company Type	0		0

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	9801	Property Store	0		0
Case G	10500	Type of Supply	0		0
Case G	10505	Calendar Setup	1	16,384	16
Case G	10550	BACS Ledger Entry	5017	545	2672
Case G	10551	BACS Register	256	384	96
Case G	10555	Fin. Charge Interest Rate	0		0
Case G	10560	Accounting Period GB	2173	192	408
Case G	10580	VAT Change Tool Setup	1	16,384	16
Case G	10581	VAT Prod. Posting Group Conv.	1	16,384	16
Case G	10582	Gen. Prod. Posting Group Conv.	0		0
Case G	50000	Budget Profile	0		0
Case G	50001	Budget/Forecast Users	0		0
Case G	50002	Document Logging Setup	1	16,384	16
Case G	50003	Forecast to Budget Buffer	0		0
Case G	50004	Pack Type	2	8,192	16
Case G	50005	RM Item Ledger Entry	0		0
Case G	50006	Pack Sub-Type	7	2,341	16
Case G	50007	Prepared	2	8,192	16
Case G	50008	Crop Type	4	4,096	16
Case G	50009	Item Cost Adjustment	1138	187	208
Case G	50010	Master No. Series	9	1,820	16
Case G	2000000002	User	2	8,192	16
Case G	2000000003	Member Of	2	8,192	16

Case Ref	Entity No.	Entity Name	No. of Records	Record Size	Size (KB)
Case G	2000000004	User Role	215	229	48
Case G	2000000005	Permission	11890	332	3856
Case G	2000000006	Company	1	16,384	16
Case G	2000000053	Windows Access Control	5	37,683	184
Case G	2000000054	Windows Login	5	16,384	80
Case G	2000000061	User Menu Level	1749	183	312
Case G	2000000065	Send-To Program	2	8,192	16
Case G	2000000066	Style Sheet	7	4,681	32
Case G	2000000067	User Default Style Sheet	2	8,192	16
Case G	2000000068	Record Link	0		80
Case G	2000000069	Client Add-in	0		0
Case G	2000000071	Object Metadata	0		0
Case G	2000000072	Profile	0		0
Case G	2000000073	User Personalization	0		0
Case G	2000000074	Profile Metadata	0		0
Case G	2000000075	User Metadata	0		0
Case G	2000000076	Web Service	0		0
Case G	2000000078	Chart	0		0
Case G	2000000079	Object Tracking	0		0
Case G	2000000080	Page Data Personalization	0		0
Case G	2000000203	Database Key Groups	28	585	16

As previously indicated, on account of ease of access, through the author's professional background, the datasets are derived from organisations within the food processing sector and each having Navision ERP systems with an SQL database back-end. Whilst this does limit the cross-system element of the samples used, the concepts, methods and frameworks described herein are developed as system-independent, and would apply to a wide cross-section of business sectors and systems.

7.4.2 Merging the Entities within the Datasets

The data in some of the cases was comprised of multiple companies within the same business. This was due to some of the source organisations having more than one company transacting over a single ERP instance.

This meant that the overall complexity of that business was fragmented over several entities, and to an extent, elevated because of it. As detailed in Table 7, Case A represented three companies within one business, stored within one raw data-file. In this case, each of these sub-companies made common use of the same ERP system, and related supporting processes.

To facilitate the application of these datasets to the framework, these dataset subsets within a case reference were merged together to consolidate the overall business as one whole. This consolidation process was managed within the **Case#_mergedata.xls** filesets.

Additionally, this step within the process was used to exclude any empty data entities. Empty data entities implied functions available within legacy systems, but not in-use within that business. Table 9 shows example results from this data merge exercise, for the same CASE G extract. These merged datasets informed the next step of the process for the normalisation of this data.

Table 9: CaseG_mergedata.xls Extract

Case Ref	Table No.	Table Name	Sum of No. of Records	Average of Record Size (B)	Sum of Size (KB)
Case G	5603	FA Setup	1	16384	16
Case G	5619	FA Journal Template	2	8192	16
Case G	5620	FA Journal Batch	2	8192	16
Case G	5622	FA Reclass. Journal Template	1	16384	16
Case G	5623	FA Reclass. Journal Batch	1	16384	16
Case G	5700	Stockkeeping Unit	27	2427	64
Case G	5717	Item Cross Reference	266	1509	392
Case G	5718	Nonstock Item	1	65536	64
Case G	5719	Nonstock Item Setup	1	16384	16
Case G	5722	Item Category	15	1092	16
Case G	5723	Product Group	9	1820	16
Case G	5744	Transfer Shipment Header	2595	218	552
Case G	5745	Transfer Shipment Line	2596	426	1080
Case G	5746	Transfer Receipt Header	2585	193	488
Case G	5747	Transfer Receipt Line	2586	472	1192
Case G	5748	Inventory Comment Line	1	16384	16
Case G	5765	Warehouse Request	14554	205	2912
Case G	5766	Warehouse Activity Header	3	16384	48
Case G	5768	Whse. Cross-Dock Opportunity	88	1303	112
Case G	5769	Warehouse Setup	1	16384	16
Case G	5772	Registered Whse. Activity Hdr.	25705	218	5472
Case G	5773	Registered Whse. Activity Line	904191	483	426768
Case G	5790	Shipping Agent Services	19	862	16

Case Ref	Table No.	Table Name	Sum of No. of Records	Average of Record Size (B)	Sum of Size (KB)
Case G	5800	Item Charge	2	24576	48
Case G	5802	Value Entry	10402133	1230	12493320
Case G	5804	Avg. Cost Adjmt. Entry Point	247444	100	24104
Case G	5813	Inventory Posting Setup	117	140	16
Case G	5814	Inventory Period	46	356	16
Case G	5815	Inventory Period Entry	35	468	16
Case G	5823	G/L - Item Ledger Relation	21916774	56	1208048
Case G	5832	Capacity Ledger Entry	909283	454	403560
Case G	5840	Standard Cost Worksheet Name	1	16384	16
Case G	5911	Service Mgt. Setup	1	16384	16
Case G	6502	Item Tracking Code	3	5461	16
Case G	6505	Lot No. Information	285694	454	126648
Case G	6506	Item Tracking Comment	6110	106	632
Case G	6507	Item Entry Relation	366958	165	58992
Case G	6508	Value Entry Relation	357362	123	42848
Case G	6509	Whse. Item Entry Relation	27446	143	3832
Case G	6550	Whse. Item Tracking Line	234	945	216
Case G	6635	Return Reason	1	16384	16
Case G	6650	Return Shipment Header	419	978	400
Case G	6651	Return Shipment Line	859	1001	840
Case G	6660	Return Receipt Header	1016	863	856
Case G	6661	Return Receipt Line	3255	921	2928
Case G	6800	Employee Portal Setup	1	16384	16

Case Ref	Table No.	Table Name	Sum of No. of Records	Average of Record Size (B)	Sum of Size (KB)
Case G	7002	Sales Price	1195	446	520
Case G	7012	Purchase Price	3	10923	32
Case G	7112	Analysis Line Template	3	5461	16
Case G	7116	Analysis Column Template	3	5461	16
Case G	7132	Item Budget Name	2	8192	16
Case G	7152	Item Analysis View	3	5461	16
Case G	7154	Item Analysis View Entry	8135	227	1800
Case G	7159	Analysis Selected Dimension	2	16384	32
Case G	7300	Zone	5	3277	16
Case G	7301	Warehouse Employee	449	292	128
Case G	7302	Bin Content	857	1434	1200
Case G	7303	Bin Type	4	4096	16
Case G	7304	Warehouse Class	2	8192	16
Case G	7307	Put-away Template Header	2	8192	16
Case G	7308	Put-away Template Line	5	3277	16
Case G	7309	Warehouse Journal Template	3	5461	16
Case G	7310	Warehouse Journal Batch	3	10923	32
Case G	7311	Warehouse Journal Line	2	94208	184
Case G	7312	Warehouse Entry	9674973	578	5464768
Case G	7313	Warehouse Register	4866798	79	375744
Case G	7316	Warehouse Receipt Header	6973	177	1208
Case G	7317	Warehouse Receipt Line	2203	718	1544
Case G	7318	Posted Whse. Receipt Header	26039	227	5776

Case Ref	Table No.	Table Name	Sum of No. of Records	Average of Record Size (B)	Sum of Size (KB)
Case G	7319	Posted Whse. Receipt Line	29659	539	15608
Case G	7320	Warehouse Shipment Header	980	276	264
Case G	7321	Warehouse Shipment Line	644	1183	744
Case G	7322	Posted Whse. Shipment Header	31668	257	7960
Case G	7323	Posted Whse. Shipment Line	207467	479	96968
Case G	7324	Whse. Put-away Request	36	455	16
Case G	7325	Whse. Pick Request	3577	69	240
Case G	7327	Whse. Worksheet Name	4	4096	16
Case G	7328	Whse. Worksheet Template	3	5461	16
Case G	7336	Bin Creation Wksh. Template	1	16384	16
Case G	7337	Bin Creation Wksh. Name	1	16384	16
Case G	7354	Bin	113	870	96
Case G	8000	Notification Setup	1	16384	16
Case G	8613	Migration Table	1	16384	16
Case G	8616	Migration Table Field	153	107	16
Case G	10505	Calendar Setup	1	16384	16
Case G	10550	BACS Ledger Entry	5017	545	2672
Case G	10551	BACS Register	256	384	96
Case G	10560	Accounting Period GB	2173	192	408
Case G	10580	VAT Change Tool Setup	1	16384	16
Case G	10581	VAT Prod. Posting Group Conv.	1	16384	16
Case G	50002	Document Logging Setup	1	16384	16
Case G	50004	Pack Type	2	8192	16

Case Ref	Table No.	Table Name	Sum of No. of Records	Average of Record Size (B)	Sum of Size (KB)
Case G	50006	Pack Sub-Type	7	2341	16
Case G	50007	Prepared	2	8192	16
Case G	50008	Crop Type	4	4096	16
Case G	50009	Item Cost Adjustment	1138	187	208
Case G	50010	Master No. Series	9	1820	16
Case G	2000000002	User	2	8192	16
Case G	2000000003	Member Of	2	8192	16
Case G	2000000004	User Role	215	229	48
Case G	2000000005	Permission	11890	332	3856
Case G	2000000006	Company	1	16384	16
Case G	2000000053	Windows Access Control	5	37683	184
Case G	2000000054	Windows Login	5	16384	80
Case G	2000000061	User Menu Level	1749	183	312
Case G	2000000065	Send-To Program	2	8192	16
Case G	2000000066	Style Sheet	7	4681	32
Case G	2000000067	User Default Style Sheet	2	8192	16
Case G	2000000203	Database Key Groups	28	585	16

7.4.3 Normalising the Datasets

The datasets for the nine business cases had various date range intervals for the data extracted. The next step within the analysis process was therefore to normalise these, such that each case dataset represented a comparative time interval. The rationale behind this process was that the larger the time interval, the bigger the data footprint, quantified in number of records and data size. Without this step, results could be skewed for those datasets with larger time intervals. This data normalisation process was carried out and stored using the **Case#_normdata.xls** filesets.

Table 10: Case Dataset Interval Normalisation

Case Reference	Start date	End date	Time Interval / Days	Time Interval / Months
Case A-1	16/09/2010	22/12/2010	97	3
Case A-2	31/12/2010	02/08/2012	580	20
Case B-1	30/04/2012	02/04/2014	702	24
Case B-2	05/12/2011	26/03/2014	842	27
Case C-1	05/03/2013	11/12/2013	281	9
Case C-2	04/04/2012	07/04/2014	733	24
Case C-3	09/07/2013	13/09/2013	66	2
Case D	30/11/2010	07/02/2014	1,165	39
Case E-1	25/03/2013	10/06/2013	77	3
Case E-2	13/04/2012	12/06/2012	60	2
Case E-3	02/10/2012	07/02/2013	128	4
Case F-1	17/01/2013	23/10/2013	279	9
Case F-2	22/04/2013	12/11/2013	204	7
Case F-3	12/10/2012	18/06/2013	249	8
Case G	26/06/2009	12/11/2010	504	17
Case H	29/04/2010	24/02/2013	1,032	34
Case I	29/11/2012	23/01/2014	420	14

For each of these business case datasets, Table 10 shows the start-date of the first transactional record and the end-date derived from the last record, within the same dataset. The number extension within the case reference indicates sub-companies of appropriate complexity within each dataset. The transactional data in this exercise referenced General Ledger postings directly for this evaluation. This was deemed to be the most accurate representation for usage across a date interval, since a business system at its core is a financial management system.

Once the transactional start-date and end-date were retrieved for each dataset, the date interval could be calculated as a factor of both days and months. The “No. of records” field of the datasets was then normalised to represent one month, for each case. From this process, the research concluded that smaller businesses would get a more accurate view of complexity by normalising their data sample for a longer time-period. This is due to the possibility of low incidence in usage for some of the process functional elements, which could therefore impact the complexity calculation.

Table 11 presents an example of a normalised extract of the dataset, using the same CASE G, which has a time interval of 17 months. The calculations applied were as follows:

i) $\text{‘No. of Records’} / \text{Interval (Months)}$

Example: Entity 6505 = 285,694 / 17 = 16,805.53

ii) $\text{‘Size (KB)’} / \text{Interval (Months)}$

Example: Entity 6505 = 126,648 / 17 = 7,449.88

Table 11: CaseG_normdata.xls Extract

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month
Case G	5603	FA Setup	0.06	16384.00	0.94
Case G	5619	FA Journal Template	0.12	8192.00	0.94
Case G	5620	FA Journal Batch	0.12	8192.00	0.94
Case G	5622	FA Reclass. Journal Template	0.06	16384.00	0.94
Case G	5623	FA Reclass. Journal Batch	0.06	16384.00	0.94
Case G	5700	Stockkeeping Unit	1.59	2427.00	3.76
Case G	5717	Item Cross Reference	15.65	1509.00	23.06
Case G	5718	Nonstock Item	0.06	65536.00	3.76

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month
Case G	5719	Nonstock Item Setup	0.06	16384.00	0.94
Case G	5722	Item Category	0.88	1092.00	0.94
Case G	5723	Product Group	0.53	1820.00	0.94
Case G	5744	Transfer Shipment Header	152.65	218.00	32.47
Case G	5745	Transfer Shipment Line	152.71	426.00	63.53
Case G	5746	Transfer Receipt Header	152.06	193.00	28.71
Case G	5747	Transfer Receipt Line	152.12	472.00	70.12
Case G	5748	Inventory Comment Line	0.06	16384.00	0.94
Case G	5765	Warehouse Request	856.12	205.00	171.29
Case G	5766	Warehouse Activity Header	0.18	16384.00	2.82
Case G	5768	Whse. Cross-Dock Opportunity	5.18	1303.00	6.59
Case G	5769	Warehouse Setup	0.06	16384.00	0.94
Case G	5772	Registered Whse. Activity Hdr.	1512.06	218.00	321.88
Case G	5773	Registered Whse. Activity Line	53187.71	483.00	25104.00
Case G	5790	Shipping Agent Services	1.12	862.00	0.94
Case G	5800	Item Charge	0.12	24576.00	2.82
Case G	5802	Value Entry	611890.18	1230.00	734901.18
Case G	5804	Avg. Cost Adjmt. Entry Point	14555.53	100.00	1417.88
Case G	5813	Inventory Posting Setup	6.88	140.00	0.94
Case G	5814	Inventory Period	2.71	356.00	0.94
Case G	5815	Inventory Period Entry	2.06	468.00	0.94
Case G	5823	G/L - Item Ledger Relation	1289222.00	56.00	71061.65
Case G	5832	Capacity Ledger Entry	53487.24	454.00	23738.82

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month
Case G	5840	Standard Cost Worksheet Name	0.06	16384.00	0.94
Case G	5911	Service Mgt. Setup	0.06	16384.00	0.94
Case G	6502	Item Tracking Code	0.18	5461.00	0.94
Case G	6505	Lot No. Information	16805.53	454.00	7449.88
Case G	6506	Item Tracking Comment	359.41	106.00	37.18
Case G	6507	Item Entry Relation	21585.76	165.00	3470.12
Case G	6508	Value Entry Relation	21021.29	123.00	2520.47
Case G	6509	Whse. Item Entry Relation	1614.47	143.00	225.41
Case G	6550	Whse. Item Tracking Line	13.76	945.00	12.71
Case G	6635	Return Reason	0.06	16384.00	0.94
Case G	6650	Return Shipment Header	24.65	978.00	23.53
Case G	6651	Return Shipment Line	50.53	1001.00	49.41
Case G	6660	Return Receipt Header	59.76	863.00	50.35
Case G	6661	Return Receipt Line	191.47	921.00	172.24
Case G	6800	Employee Portal Setup	0.06	16384.00	0.94
Case G	7002	Sales Price	70.29	446.00	30.59
Case G	7012	Purchase Price	0.18	10923.00	1.88
Case G	7112	Analysis Line Template	0.18	5461.00	0.94
Case G	7116	Analysis Column Template	0.18	5461.00	0.94
Case G	7132	Item Budget Name	0.12	8192.00	0.94
Case G	7152	Item Analysis View	0.18	5461.00	0.94
Case G	7154	Item Analysis View Entry	478.53	227.00	105.88
Case G	7159	Analysis Selected Dimension	0.12	16384.00	1.88

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month
Case G	7300	Zone	0.29	3277.00	0.94
Case G	7301	Warehouse Employee	26.41	292.00	7.53
Case G	7302	Bin Content	50.41	1434.00	70.59
Case G	7303	Bin Type	0.24	4096.00	0.94
Case G	7304	Warehouse Class	0.12	8192.00	0.94
Case G	7307	Put-away Template Header	0.12	8192.00	0.94
Case G	7308	Put-away Template Line	0.29	3277.00	0.94
Case G	7309	Warehouse Journal Template	0.18	5461.00	0.94
Case G	7310	Warehouse Journal Batch	0.18	10923.00	1.88
Case G	7311	Warehouse Journal Line	0.12	94208.00	10.82
Case G	7312	Warehouse Entry	569116.06	578.00	321456.94
Case G	7313	Warehouse Register	286282.24	79.00	22102.59
Case G	7316	Warehouse Receipt Header	410.18	177.00	71.06
Case G	7317	Warehouse Receipt Line	129.59	718.00	90.82
Case G	7318	Posted Whse. Receipt Header	1531.71	227.00	339.76
Case G	7319	Posted Whse. Receipt Line	1744.65	539.00	918.12
Case G	7320	Warehouse Shipment Header	57.65	276.00	15.53
Case G	7321	Warehouse Shipment Line	37.88	1183.00	43.76
Case G	7322	Posted Whse. Shipment Header	1862.82	257.00	468.24
Case G	7323	Posted Whse. Shipment Line	12203.94	479.00	5704.00
Case G	7324	Whse. Put-away Request	2.12	455.00	0.94
Case G	7325	Whse. Pick Request	210.41	69.00	14.12
Case G	7327	Whse. Worksheet Name	0.24	4096.00	0.94

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month
Case G	7328	Whse. Worksheet Template	0.18	5461.00	0.94
Case G	7336	Bin Creation Wksh. Template	0.06	16384.00	0.94
Case G	7337	Bin Creation Wksh. Name	0.06	16384.00	0.94
Case G	7354	Bin	6.65	870.00	5.65
Case G	8000	Notification Setup	0.06	16384.00	0.94
Case G	8613	Migration Table	0.06	16384.00	0.94
Case G	8616	Migration Table Field	9.00	107.00	0.94
Case G	10505	Calendar Setup	0.06	16384.00	0.94
Case G	10550	BACS Ledger Entry	295.12	545.00	157.18
Case G	10551	BACS Register	15.06	384.00	5.65
Case G	10560	Accounting Period GB	127.82	192.00	24.00
Case G	10580	VAT Change Tool Setup	0.06	16384.00	0.94
Case G	10581	VAT Prod. Posting Group Conv.	0.06	16384.00	0.94
Case G	50002	Document Logging Setup	0.06	16384.00	0.94
Case G	50004	Pack Type	0.12	8192.00	0.94
Case G	50006	Pack Sub-Type	0.41	2341.00	0.94
Case G	50007	Prepared	0.12	8192.00	0.94
Case G	50008	Crop Type	0.24	4096.00	0.94
Case G	50009	Item Cost Adjustment	66.94	187.00	12.24
Case G	50010	Master No. Series	9.00	1820.00	16.00
Case G	2000000002	User	2.00	8192.00	16.00
Case G	2000000003	Member Of	2.00	8192.00	16.00
Case G	2000000004	User Role	215.00	229.00	48.00

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month
Case G	2000000005	Permission	11890.00	332.00	3856.00
Case G	2000000006	Company	1.00	16384.00	16.00
Case G	2000000053	Windows Access Control	5.00	37683.00	184.00
Case G	2000000054	Windows Login	5.00	16384.00	80.00
Case G	2000000061	User Menu Level	1749.00	183.00	312.00
Case G	2000000065	Send-To Program	2.00	8192.00	16.00
Case G	2000000066	Style Sheet	7.00	4681.00	32.00
Case G	2000000067	User Default Style Sheet	2.00	8192.00	16.00
Case G	2000000203	Database Key Groups	28.00	585.00	16.00

Business data is typically not all date dependant however, with the key type of data evident in the datasets classified as:

- **Reference (Standing) data** – sets of records that describe entities within the business, set up for long-term use. Updates of this data is date independent, example Customer records.
- **Transactional data** – this is data generated through business operations, such as financial postings. This data always has a time element, and is the fastest growing part of the dataset over a time-period.

For additional accuracy within the normalisation process, only transactional data has been normalised. This is due to Reference Data updates being independent of time periods. As an example, one Customer reference record could have hundreds of related orders created monthly. This analysis therefore makes a subjective separation of the case datasets between Reference and Transactional data, using the inferred notion that an entity name having Ledger or Entry within its caption implying transactional records.

7.4.4 SCF Unified Model – Process Node Mapping

The first application of the SCFUM model involved the mapping of each of the dataset entities to the process nodes within the model. This was achieved by understanding the context of each specific entity within the wider business process, and then mapping this to the equivalent SCFUM model process node. This was done using pivot tables, and a manual iterative review of the data entities within the **Case#_nodemap.xls** filesets.

Pivot tables are an example of Business Intelligence (BI). BI is general term that refers to a variety of software applications used to analyse an organisation's raw data. BI as a discipline is made up of several related activities, including data mining, analytical processing, querying and reporting (Mulcahy, 2007).

Table 12 represents an example of the results of this mapping process for the same CASE G extract. As an example, taking business entity no. 6505: “Lot No. Information”, the author infers that this is related to the warehouse process within the organisation; which process is represented by the Warehouse Process Node within the SCFUM model. This mapping process was iteratively repeated for each of the nine business cases. The “Entities with Records” column within Table 15 highlights the number of data entities that were individually analysed for each of the business cases.

Table 12: CaseG_nodemap.xls Extract

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node
Case G	5603	FA Setup	0.06	16384.00	0.94	Finance
Case G	5619	FA Journal Template	0.12	8192.00	0.94	Finance
Case G	5620	FA Journal Batch	0.12	8192.00	0.94	Finance
Case G	5622	FA Reclass. Journal Template	0.06	16384.00	0.94	Finance
Case G	5623	FA Reclass. Journal Batch	0.06	16384.00	0.94	Finance

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node
Case G	5700	Stockkeeping Unit	1.59	2427.00	3.76	Warehouse
Case G	5717	Item Cross Reference	15.65	1509.00	23.06	Sales
Case G	5718	Nonstock Item	0.06	65536.00	3.76	Sales / Procurement
Case G	5719	Nonstock Item Setup	0.06	16384.00	0.94	Sales / Procurement
Case G	5722	Item Category	0.88	1092.00	0.94	Warehouse
Case G	5723	Product Group	0.53	1820.00	0.94	Warehouse
Case G	5744	Transfer Shipment Header	152.65	218.00	32.47	Warehouse
Case G	5745	Transfer Shipment Line	152.71	426.00	63.53	Warehouse
Case G	5746	Transfer Receipt Header	152.06	193.00	28.71	Warehouse
Case G	5747	Transfer Receipt Line	152.12	472.00	70.12	Warehouse
Case G	5748	Inventory Comment Line	0.06	16384.00	0.94	Warehouse
Case G	5765	Warehouse Request	856.12	205.00	171.29	Warehouse
Case G	5766	Warehouse Activity Header	0.18	16384.00	2.82	Warehouse
Case G	5768	Whse. Cross-Dock Opportunity	5.18	1303.00	6.59	Warehouse
Case G	5769	Warehouse Setup	0.06	16384.00	0.94	Warehouse
Case G	5772	Registered Whse. Activity Hdr.	1512.06	218.00	321.88	Warehouse
Case G	5773	Registered Whse. Activity Line	53187.71	483.00	25104.00	Warehouse

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node
Case G	5790	Shipping Agent Services	1.12	862.00	0.94	Logistics
Case G	5800	Item Charge	0.12	24576.00	2.82	Finance
Case G	5802	Value Entry	611890.18	1230.00	734901.18	Finance
Case G	5804	Avg. Cost Adjmt. Entry Point	14555.53	100.00	1417.88	Finance
Case G	5813	Inventory Posting Setup	6.88	140.00	0.94	Finance
Case G	5814	Inventory Period	2.71	356.00	0.94	Warehouse
Case G	5815	Inventory Period Entry	2.06	468.00	0.94	Warehouse
Case G	5823	G/L - Item Ledger Relation	1289222.00	56.00	71061.65	Finance
Case G	5832	Capacity Ledger Entry	53487.24	454.00	23738.82	Production
Case G	5840	Standard Cost Worksheet Name	0.06	16384.00	0.94	Warehouse
Case G	5911	Service Mgt. Setup	0.06	16384.00	0.94	Services
Case G	6502	Item Tracking Code	0.18	5461.00	0.94	Warehouse
Case G	6505	Lot No. Information	16805.53	454.00	7449.88	Warehouse
Case G	6506	Item Tracking Comment	359.41	106.00	37.18	Warehouse
Case G	6507	Item Entry Relation	21585.76	165.00	3470.12	Warehouse
Case G	6508	Value Entry Relation	21021.29	123.00	2520.47	Warehouse
Case G	6509	Whse. Item Entry Relation	1614.47	143.00	225.41	Warehouse

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node
Case G	6550	Whse. Item Tracking Line	13.76	945.00	12.71	Warehouse
Case G	6635	Return Reason	0.06	16384.00	0.94	Sales / Procurement
Case G	6650	Return Shipment Header	24.65	978.00	23.53	Sales
Case G	6651	Return Shipment Line	50.53	1001.00	49.41	Sales
Case G	6660	Return Receipt Header	59.76	863.00	50.35	Sales
Case G	6661	Return Receipt Line	191.47	921.00	172.24	Sales
Case G	6800	Employee Portal Setup	0.06	16384.00	0.94	HR
Case G	7002	Sales Price	70.29	446.00	30.59	Sales
Case G	7012	Purchase Price	0.18	10923.00	1.88	Procurement
Case G	7112	Analysis Line Template	0.18	5461.00	0.94	Finance
Case G	7116	Analysis Column Template	0.18	5461.00	0.94	Finance
Case G	7132	Item Budget Name	0.12	8192.00	0.94	Warehouse
Case G	7152	Item Analysis View	0.18	5461.00	0.94	Warehouse
Case G	7154	Item Analysis View Entry	478.53	227.00	105.88	Warehouse
Case G	7159	Analysis Selected Dimension	0.12	16384.00	1.88	Finance
Case G	7300	Zone	0.29	3277.00	0.94	Warehouse
Case G	7301	Warehouse Employee	26.41	292.00	7.53	Warehouse

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node
Case G	7302	Bin Content	50.41	1434.00	70.59	Warehouse
Case G	7303	Bin Type	0.24	4096.00	0.94	Warehouse
Case G	7304	Warehouse Class	0.12	8192.00	0.94	Warehouse
Case G	7307	Put-away Template Header	0.12	8192.00	0.94	Warehouse
Case G	7308	Put-away Template Line	0.29	3277.00	0.94	Warehouse
Case G	7309	Warehouse Journal Template	0.18	5461.00	0.94	Warehouse
Case G	7310	Warehouse Journal Batch	0.18	10923.00	1.88	Warehouse
Case G	7311	Warehouse Journal Line	0.12	94208.00	10.82	Warehouse
Case G	7312	Warehouse Entry	569116.06	578.00	321456.94	Warehouse
Case G	7313	Warehouse Register	286282.24	79.00	22102.59	Warehouse
Case G	7316	Warehouse Receipt Header	410.18	177.00	71.06	Warehouse
Case G	7317	Warehouse Receipt Line	129.59	718.00	90.82	Warehouse
Case G	7318	Posted Whse. Receipt Header	1531.71	227.00	339.76	Warehouse
Case G	7319	Posted Whse. Receipt Line	1744.65	539.00	918.12	Warehouse
Case G	7320	Warehouse Shipment Header	57.65	276.00	15.53	Warehouse
Case G	7321	Warehouse Shipment Line	37.88	1183.00	43.76	Warehouse
Case G	7322	Posted Whse. Shipment Header	1862.82	257.00	468.24	Warehouse

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node
Case G	7323	Posted Whse. Shipment Line	12203.94	479.00	5704.00	Warehouse
Case G	7324	Whse. Put-away Request	2.12	455.00	0.94	Warehouse
Case G	7325	Whse. Pick Request	210.41	69.00	14.12	Warehouse
Case G	7327	Whse. Worksheet Name	0.24	4096.00	0.94	Warehouse
Case G	7328	Whse. Worksheet Template	0.18	5461.00	0.94	Warehouse
Case G	7336	Bin Creation Wksh. Template	0.06	16384.00	0.94	Warehouse
Case G	7337	Bin Creation Wksh. Name	0.06	16384.00	0.94	Warehouse
Case G	7354	Bin	6.65	870.00	5.65	Warehouse
Case G	8000	Notification Setup	0.06	16384.00	0.94	Administration
Case G	8613	Migration Table	0.06	16384.00	0.94	Finance
Case G	8616	Migration Table Field	9.00	107.00	0.94	Finance
Case G	10505	Calendar Setup	0.06	16384.00	0.94	Finance
Case G	10550	BACS Ledger Entry	295.12	545.00	157.18	Finance
Case G	10551	BACS Register	15.06	384.00	5.65	Finance
Case G	10560	Accounting Period GB	127.82	192.00	24.00	Finance
Case G	10580	VAT Change Tool Setup	0.06	16384.00	0.94	Finance
Case G	10581	VAT Prod. Posting Group Conv.	0.06	16384.00	0.94	Finance

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node
Case G	50002	Document Logging Setup	0.06	16384.00	0.94	#N/A
Case G	50004	Pack Type	0.12	8192.00	0.94	#N/A
Case G	50006	Pack Sub-Type	0.41	2341.00	0.94	#N/A
Case G	50007	Prepared	0.12	8192.00	0.94	#N/A
Case G	50008	Crop Type	0.24	4096.00	0.94	#N/A
Case G	50009	Item Cost Adjustment	66.94	187.00	12.24	#N/A
Case G	50010	Master No. Series	9.00	1820.00	16.00	#N/A
Case G	2000000002	User	2.00	8192.00	16.00	Administration
Case G	2000000003	Member Of	2.00	8192.00	16.00	Administration
Case G	2000000004	User Role	215.00	229.00	48.00	Administration
Case G	2000000005	Permission	11890.00	332.00	3856.00	Administration
Case G	2000000006	Company	1.00	16384.00	16.00	Administration
Case G	2000000053	Windows Access Control	5.00	37683.00	184.00	Administration
Case G	2000000054	Windows Login	5.00	16384.00	80.00	Administration
Case G	2000000061	User Menu Level	1749.00	183.00	312.00	Administration
Case G	2000000065	Send-To Program	2.00	8192.00	16.00	Administration
Case G	2000000066	Style Sheet	7.00	4681.00	32.00	Administration

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node
Case G	2000000067	User Default Style Sheet	2.00	8192.00	16.00	Administration
Case G	2000000203	Database Key Groups	28.00	585.00	16.00	Administration

Since a Process Node represents a grouping of Process Functional Elements within the model, the rationale to initially analyse the datasets at the process node level was twofold:

I. Facilitate the subsequent low-level mapping of the individual entities to the model's functional elements and thus allowing analysis of these, one Process Node at a time.
II. Validate the SCFUM model for comprehensive Process Node coverage.

7.4.5 SCF Unified Model – Process Functional Element Mapping

The second part of the SCFUM model application was the mapping of the process functional elements to the case data entities using the process node grouping derived from the initial analysis, shown in Table 12. Using spreadsheet pivot tables, with their related data manipulation and filtering functions, data entities subsets were separately analysed in detail, for each process node.

Within this process, a SCFUM Process Functional element was then mapped to each case entity within the **Case#_elemmap.xls** filesets. This enabled the generation of a unified overview for the whole dataset, within each case study (and therefore for each business). A worked example from the outcome of this exercise is represented in Table 13, for the same CASE G extract.

Table 13: CaseG_elemmap.xls Extract

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node	SCF Process Element
Case G	5603	FA Setup	0.06	16384.00	0.94	Finance	Fixed Assets
Case G	5619	FA Journal Template	0.12	8192.00	0.94	Finance	FA Journals
Case G	5620	FA Journal Batch	0.12	8192.00	0.94	Finance	FA Journals
Case G	5622	FA Reclass. Journal Template	0.06	16384.00	0.94	Finance	FA Journals
Case G	5623	FA Reclass. Journal Batch	0.06	16384.00	0.94	Finance	FA Journals
Case G	5700	Stockkeeping Unit	1.59	2427.00	3.76	Warehouse	Stock Card
Case G	5717	Item Cross Reference	15.65	1509.00	23.06	Sales	Stock Card
Case G	5718	Nonstock Item	0.06	65536.00	3.76	Sales / Procurement	Stock Card
Case G	5719	Nonstock Item Setup	0.06	16384.00	0.94	Sales / Procurement	Stock Card
Case G	5722	Item Category	0.88	1092.00	0.94	Warehouse	Stock Card
Case G	5723	Product Group	0.53	1820.00	0.94	Warehouse	Stock Card
Case G	5744	Transfer Shipment Header	152.65	218.00	32.47	Warehouse	Transfer Orders
Case G	5745	Transfer Shipment Line	152.71	426.00	63.53	Warehouse	Transfer Orders
Case G	5746	Transfer Receipt Header	152.06	193.00	28.71	Warehouse	Transfer Orders
Case G	5747	Transfer Receipt Line	152.12	472.00	70.12	Warehouse	Transfer Orders
Case G	5748	Inventory Comment Line	0.06	16384.00	0.94	Warehouse	Inventory

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node	SCF Process Element
Case G	5765	Warehouse Request	856.12	205.00	171.29	Warehouse	Warehouse Movements
Case G	5766	Warehouse Activity Header	0.18	16384.00	2.82	Warehouse	Warehouse Movements
Case G	5768	Whse. Cross-Dock Opportunity	5.18	1303.00	6.59	Warehouse	Warehouse Movements
Case G	5769	Warehouse Setup	0.06	16384.00	0.94	Warehouse	Warehouse Locations
Case G	5772	Registered Whse. Activity Hdr.	1512.06	218.00	321.88	Warehouse	Warehouse Movements
Case G	5773	Registered Whse. Activity Line	53187.71	483.00	25104.00	Warehouse	Warehouse Movements
Case G	5790	Shipping Agent Services	1.12	862.00	0.94	Logistics	Transport Agents
Case G	5800	Item Charge	0.12	24576.00	2.82	Finance	General Journal
Case G	5802	Value Entry	611890.18	1230.00	734901.18	Finance	General Ledger Registers
Case G	5804	Avg. Cost Adjmt. Entry Point	14555.53	100.00	1417.88	Finance	General Ledger Registers
Case G	5813	Inventory Posting Setup	6.88	140.00	0.94	Finance	General Journal
Case G	5814	Inventory Period	2.71	356.00	0.94	Warehouse	Stock Card
Case G	5815	Inventory Period Entry	2.06	468.00	0.94	Warehouse	Inventory

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node	SCF Process Element
Case G	5823	G/L - Item Ledger Relation	1289222.00	56.00	71061.65	Finance	General Ledger Registers
Case G	5832	Capacity Ledger Entry	53487.24	454.00	23738.82	Production	Capacity Journals
Case G	5840	Standard Cost Worksheet Name	0.06	16384.00	0.94	Warehouse	Stock Card
Case G	5911	Service Mgt. Setup	0.06	16384.00	0.94	Services	Services Mgmt
Case G	6502	Item Tracking Code	0.18	5461.00	0.94	Warehouse	Stock Analysis
Case G	6505	Lot No. Information	16805.53	454.00	7449.88	Warehouse	Stock Card
Case G	6506	Item Tracking Comment	359.41	106.00	37.18	Warehouse	Stock Analysis
Case G	6507	Item Entry Relation	21585.76	165.00	3470.12	Warehouse	Inventory
Case G	6508	Value Entry Relation	21021.29	123.00	2520.47	Warehouse	Inventory
Case G	6509	Whse. Item Entry Relation	1614.47	143.00	225.41	Warehouse	Inventory
Case G	6550	Whse. Item Tracking Line	13.76	945.00	12.71	Warehouse	Stock Analysis
Case G	6635	Return Reason	0.06	16384.00	0.94	Sales / Procurement	Return Orders
Case G	6650	Return Shipment Header	24.65	978.00	23.53	Sales	Return Orders
Case G	6651	Return Shipment Line	50.53	1001.00	49.41	Sales	Return Orders
Case G	6660	Return Receipt Header	59.76	863.00	50.35	Sales	Return Orders

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node	SCF Process Element
Case G	6661	Return Receipt Line	191.47	921.00	172.24	Sales	Return Orders
Case G	6800	Employee Portal Setup	0.06	16384.00	0.94	HR	Employee Portal
Case G	7002	Sales Price	70.29	446.00	30.59	Sales	Sales Prices
Case G	7012	Purchase Price	0.18	10923.00	1.88	Procurement	Purchase Prices
Case G	7112	Analysis Line Template	0.18	5461.00	0.94	Finance	Chart of accounts
Case G	7116	Analysis Column Template	0.18	5461.00	0.94	Finance	Chart of accounts
Case G	7132	Item Budget Name	0.12	8192.00	0.94	Warehouse	Stock Analysis
Case G	7152	Item Analysis View	0.18	5461.00	0.94	Warehouse	Stock Analysis
Case G	7154	Item Analysis View Entry	478.53	227.00	105.88	Warehouse	Stock Analysis
Case G	7159	Analysis Selected Dimension	0.12	16384.00	1.88	Finance	Chart of accounts
Case G	7300	Zone	0.29	3277.00	0.94	Warehouse	Warehouse Locations
Case G	7301	Warehouse Employee	26.41	292.00	7.53	Warehouse	Warehouse Locations
Case G	7302	Bin Content	50.41	1434.00	70.59	Warehouse	Bin Contents
Case G	7303	Bin Type	0.24	4096.00	0.94	Warehouse	Bin Contents
Case G	7304	Warehouse Class	0.12	8192.00	0.94	Warehouse	Warehouse Locations
Case G	7307	Put-away Template Header	0.12	8192.00	0.94	Warehouse	Warehouse Put-aways

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node	SCF Process Element
Case G	7308	Put-away Template Line	0.29	3277.00	0.94	Warehouse	Warehouse Put-aways
Case G	7309	Warehouse Journal Template	0.18	5461.00	0.94	Warehouse	Item Journals
Case G	7310	Warehouse Journal Batch	0.18	10923.00	1.88	Warehouse	Item Journals
Case G	7311	Warehouse Journal Line	0.12	94208.00	10.82	Warehouse	Item Journals
Case G	7312	Warehouse Entry	569116.06	578.00	321456.94	Warehouse	Inventory
Case G	7313	Warehouse Register	286282.24	79.00	22102.59	Warehouse	Inventory
Case G	7316	Warehouse Receipt Header	410.18	177.00	71.06	Warehouse	Warehouse Receipts
Case G	7317	Warehouse Receipt Line	129.59	718.00	90.82	Warehouse	Warehouse Receipts
Case G	7318	Posted Whse. Receipt Header	1531.71	227.00	339.76	Warehouse	Warehouse Receipts
Case G	7319	Posted Whse. Receipt Line	1744.65	539.00	918.12	Warehouse	Warehouse Receipts
Case G	7320	Warehouse Shipment Header	57.65	276.00	15.53	Warehouse	Warehouse Shipments
Case G	7321	Warehouse Shipment Line	37.88	1183.00	43.76	Warehouse	Warehouse Shipments
Case G	7322	Posted Whse. Shipment Header	1862.82	257.00	468.24	Warehouse	Warehouse Shipments
Case G	7323	Posted Whse. Shipment Line	12203.94	479.00	5704.00	Warehouse	Warehouse Shipments
Case G	7324	Whse. Put-away	2.12	455.00	0.94	Warehouse	Warehouse

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node	SCF Process Element
		Request					Put-aways
Case G	7325	Whse. Pick Request	210.41	69.00	14.12	Warehouse	Warehouse Picks
Case G	7327	Whse. Worksheet Name	0.24	4096.00	0.94	Warehouse	Item Journals
Case G	7328	Whse. Worksheet Template	0.18	5461.00	0.94	Warehouse	Item Journals
Case G	7336	Bin Creation Wksh. Template	0.06	16384.00	0.94	Warehouse	Bin Contents
Case G	7337	Bin Creation Wksh. Name	0.06	16384.00	0.94	Warehouse	Bin Contents
Case G	7354	Bin	6.65	870.00	5.65	Warehouse	Bin Contents
Case G	8000	Notification Setup	0.06	16384.00	0.94	Administration	Administration
Case G	8613	Migration Table	0.06	16384.00	0.94	Finance	Chart of accounts
Case G	8616	Migration Table Field	9.00	107.00	0.94	Finance	Chart of accounts
Case G	10505	Calendar Setup	0.06	16384.00	0.94	Finance	Chart of accounts
Case G	10550	BACS Ledger Entry	295.12	545.00	157.18	Finance	Bank Accounts
Case G	10551	BACS Register	15.06	384.00	5.65	Finance	Bank Accounts
Case G	10560	Accounting Period GB	127.82	192.00	24.00	Finance	Chart of accounts
Case G	10580	VAT Change Tool Setup	0.06	16384.00	0.94	Finance	Tax Mgmt
Case G	10581	VAT Prod. Posting	0.06	16384.00	0.94	Finance	Tax Mgmt

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node	SCF Process Element
		Group Conv.					
Case G	50002	Document Logging Setup	0.06	16384.00	0.94	#N/A	#N/A
Case G	50004	Pack Type	0.12	8192.00	0.94	#N/A	#N/A
Case G	50006	Pack Sub-Type	0.41	2341.00	0.94	#N/A	#N/A
Case G	50007	Prepared	0.12	8192.00	0.94	#N/A	#N/A
Case G	50008	Crop Type	0.24	4096.00	0.94	#N/A	#N/A
Case G	50009	Item Cost Adjustment	66.94	187.00	12.24	#N/A	#N/A
Case G	50010	Master No. Series	9.00	1820.00	16.00	#N/A	#N/A
Case G	2000000002	User	2.00	8192.00	16.00	Administration	Administration
Case G	2000000003	Member Of	2.00	8192.00	16.00	Administration	Administration
Case G	2000000004	User Role	215.00	229.00	48.00	Administration	Administration
Case G	2000000005	Permission	11890.00	332.00	3856.00	Administration	Administration
Case G	2000000006	Company	1.00	16384.00	16.00	Administration	Administration
Case G	2000000053	Windows Access Control	5.00	37683.00	184.00	Administration	Administration
Case G	2000000054	Windows Login	5.00	16384.00	80.00	Administration	Administration
Case G	2000000061	User Menu Level	1749.00	183.00	312.00	Administration	Administration
Case G	2000000065	Send-To Program	2.00	8192.00	16.00	Administration	Administration
Case G	2000000066	Style Sheet	7.00	4681.00	32.00	Administration	Administration
Case G	2000000067	User Default Style Sheet	2.00	8192.00	16.00	Administration	Administration
Case G	2000000203	Database Key Groups	28.00	585.00	16.00	Administration	Administration

Within this same mapping process, analysis of the resultant data derived several instances where the case entities could not be mapped. These entities were marked as "No value available" entries, represented as "N/A", within the mapping result datasets. Detailed analysis of these entries yielded two key outputs:

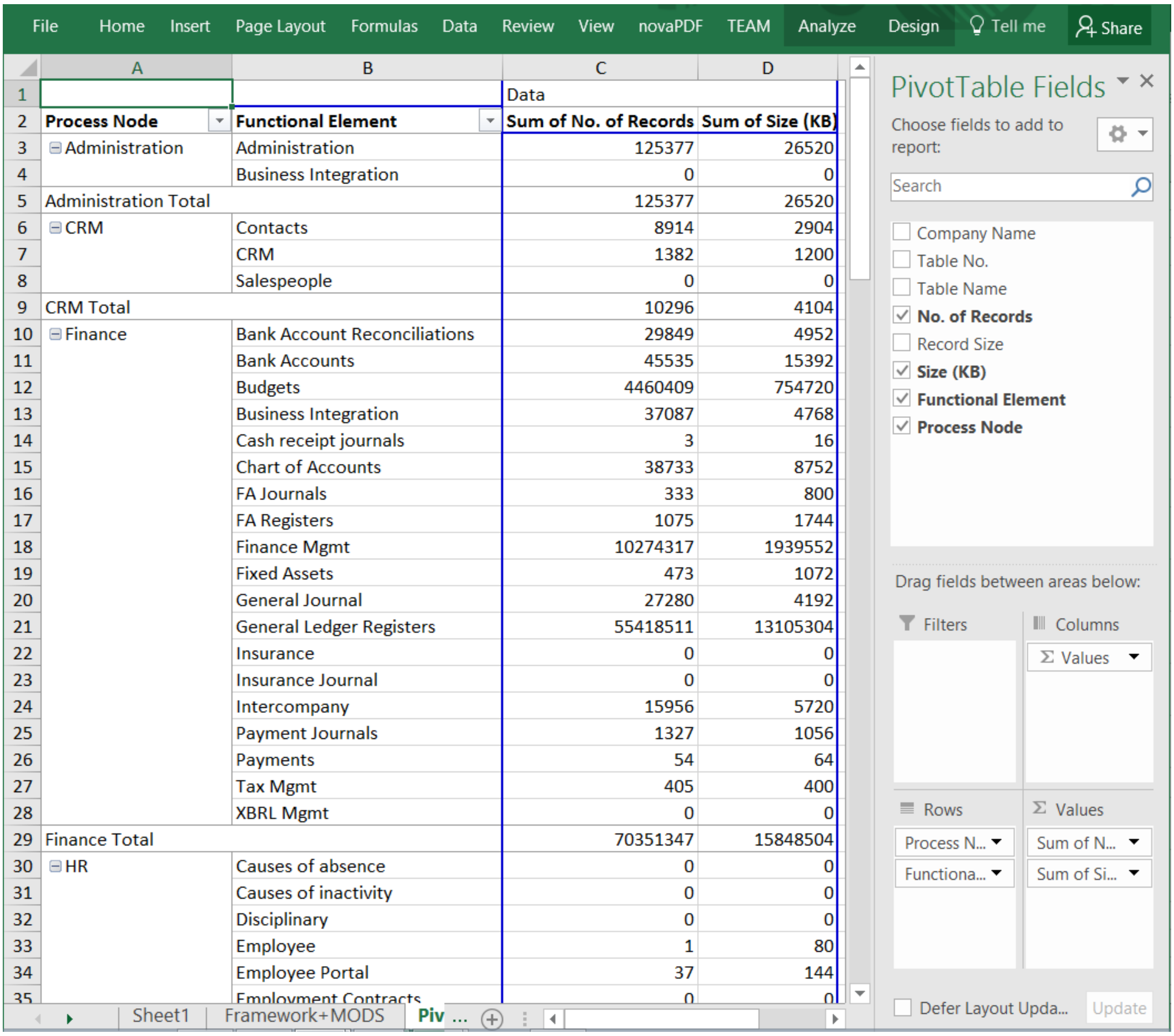
1. enabling the calibration of the SCFUM model within the iterations where required.
2. allowing the identification of business specific customisations that would otherwise be required within a solution deployment.

Reasons for these mapping variances are explained later in this section. Table 14 shows an example of such variances between the initial version of the SCFUM model and the CASE G dataset extract.

Table 14: Mapping Variances from SCFUMv1. (Case G Extract)

Case Ref	Entity No.	Entity Name	Sum of No. of Records / Month	Average of Record Size (B)	Sum of Size (KB) / Month	SCF Process Node	SCF Process Element
Case G	50002	Document Logging Setup	0.06	16384	0.94	#N/A	#N/A
Case G	50004	Pack Type	0.12	8192	0.94	#N/A	#N/A
Case G	50006	Pack Sub-Type	0.41	2341	0.94	#N/A	#N/A
Case G	50007	Prepared	0.12	8192	0.94	#N/A	#N/A
Case G	50008	Crop Type	0.24	4096	0.94	#N/A	#N/A
Case G	50009	Item Cost Adjustment	66.94	187	12.24	#N/A	#N/A
Case G	50010	Master No. Series	9	1820	16	#N/A	#N/A

Following the mapping of the SCFUM model to each business case, the next step in the SCF framework application process was the iterative analysis and interpretation of these results. Pivot tables were manipulated in spreadsheets to facilitate these analyses, with the resultant datasets stored in the file **Process_Node_Mapping_Stats.xls**. An example of such a pivot table is shown in Figure 29.



	Data	
Process Node	Functional Element	Sum of No. of Records	Sum of Size (KB)
⊟Administration	Administration	125377	26520
	Business Integration	0	0
Administration Total		125377	26520
⊟CRM	Contacts	8914	2904
	CRM	1382	1200
	Salespeople	0	0
CRM Total		10296	4104
⊟Finance	Bank Account Reconciliations	29849	4952
	Bank Accounts	45535	15392
	Budgets	4460409	754720
	Business Integration	37087	4768
	Cash receipt journals	3	16
	Chart of Accounts	38733	8752
	FA Journals	333	800
	FA Registers	1075	1744
	Finance Mgmt	10274317	1939552
	Fixed Assets	473	1072
	General Journal	27280	4192
	General Ledger Registers	55418511	13105304
	Insurance	0	0
	Insurance Journal	0	0
	Intercompany	15956	5720
	Payment Journals	1327	1056
	Payments	54	64
	Tax Mgmt	405	400
	XBRL Mgmt	0	0
Finance Total		70351347	15848504
⊟HR	Causes of absence	0	0
	Causes of inactivity	0	0
	Disciplinary	0	0
	Employee	1	80
	Employee Portal	37	144
	Employment Contracts	0	0

Figure 29: Extract of Dataset Analyses using Pivot Tables

The analyses and results yielded by the case study datasets following this SCF framework application is summarised in Table 15.

Table 15: Process Node Analyses Results SCFUMv1

Case Ref	Entities with Records	Merged Entities	(N/A) Non Mapped Entities	(N/A) Non mapped Entities %	Business Specific Entities	Business Specific N/A Entities w/Rec	BSE N/A Var %	Industry Specific N/A Entities	ISE N/A Var %
Case A	1,676	876	285	33%	61	41	14%	244	86%
Case B	2,119	2,119	297	14%	100	58	20%	239	80%
Case C	2,685	623	278	45%	146	90	32%	188	68%
Case D	814	544	237	44%	74	65	27%	172	73%
Case E	3,143	566	260	46%	75	52	20%	208	80%
Case F	2,787	865	272	31%	44	28	10%	244	90%
Case G	543	543	251	46%	103	55	22%	196	78%
Case H	497	471	197	42%	104	27	14%	170	86%
Case I	371	371	98	26%	69	31	32%	67	68%

This data analysis used the following key calculations to interpret the SCFUM model application results. These were output to the data file **Process_Node_Mapping_Stats.xls**:

- **Case Ref:** The business case study reference anonymously representing a business dataset.

- **Entities with Records:** This is the total number of case entities (CE) within the raw dataset containing records.

$$\sum_{CE=1}^{n} Case\ Entities\,, CE \neq \Phi$$

- **Merged Entities:** The total number of merged case entities (MCE), where the raw dataset covered multiple companies within the same business.

$$\sum_{MCE=1}^{n} Merged\ Case\ Entities\,, MCE \neq \Phi$$

- **(N/A) Non-Mapped Entities:** The number of case entities which had no initial representation within any SCF Process Element in the SCFUM model, and therefore could not be immediately mapped.

$$\sum (N/A)\ Non\ Mapped\ Entities$$

- **(N/A) Non-Mapped Entities %:** A calculation of the derived variance between the SCFUM model and the business case, as a percentage of merged case entities.

$$\left(\frac{(N/A)\ Non\text{-}Mapped\ Entities}{Merged\ Case\ Entities}\right) \times 100$$

- **Business Specific Entities:** The number of entities representing modifications and custom functionality specific for that business. These could be comprised of both general process elements and “no value available” (N/A) business specific processes. These entities therefore had to be analysed separately for inclusion or exclusion from the SCFUM model. An example of exclusion entity mapping is represented in Table 16.

$$\sum Business\ Specific\ Entities$$

- **Business Specific N/A Entities w/Rec:** The number of business specific N/A entities (BSNE) representing modifications and custom functionality specific for that business that contain records, indicating actual use.

$$\sum_{BSNE=1}^{n} Business\ Specific\ N/A\ Entities\ \ , BSNE \neq \ \Phi$$

- **BSE N/A Var %:** The calculated variance for business specific entities with no applicable mapping with respect to the sum of all the non-mapped entities.

$$\left(\frac{\text{“BSE N/A Entities w/rec”}}{\text{”(N/A) Non Mapped Entities”}}\right) \times 100$$

- **Industry Specific N/A Entities:** The total number of non-mapped entities representing Industry specific functionality, common across businesses within that same industry.

$$\text{“}(N/A)\ Non\ Mapped\ Entities\text{”}\ - \text{“}Business\ Specific\ N/A\ Entities\ w/rec\text{”}$$

- **Industry Specific Entities (ISE) N/A Var %:** The calculated variance for Industry specific entities with no applicable mapping with respect to the sum of all the non-mapped entities.

$$\left(\frac{\text{ISE N/A Entities}}{\text{(N/A) Non Mapped Entities}}\right) \times 100$$

7.5 Analysis of “No Applicable Values” within Results

From this analysis, entities within the case datasets that could not be mapped and other incongruences within the mapping process allowed review of the model, and fine-tuning of the process and functional elements described therein. An extract of these “No Applicable Value” (N/A) results for CASE G is shown in Table 16. In depth review of these entities ensured better coverage for such variances in future applications of the model. The resultant datasets were processed in file **NAV_VAR_SCFMap.xls**.

The general cause for these SCFUM model mapping variances were identified as follows:

1) **Data entities having no mapping within the framework** - Some business data entities did not fit within the initial model and therefore could not be mapped.
2) **Accuracy of the mapping process** - Some case study data entities could not be accurately identified within a generic business process context and therefore could not be accurately mapped within the framework.
3) **Inefficiencies in mapping process** -The initial mapping process attempted a direct mapping of data entities to the low-level process functional elements of the model. This was time-consuming and considered an inefficient approach when applied to the actual business case data.

An in-depth evaluation of points 1) and 2) above suggest the higher number of differences presented in Table 15 resulted from:

a) Business case specific, custom process functional elements.

Individual businesses extend and modify their process functional elements to satisfy specific requirements within their operations. These resulted in deviations from the SCFUM model. These deviations still had to be factored into the metric calculations to derive an accurate SCI for the business during the analysis process, as shown in Table 16. Due to these being business specific however, they have not been included within the SCFUM model.

Application of the framework would in effect likely result in a variance for any business specific process functional elements. In reality, these variances would still be an invaluable factor in calculating an estimate, both for the level of complexity in a business and to provide an indication of the level of modification that that business would require.

b) Industry specific process functional elements.

Different industries might have specific process functional elements that are not evident in other sectors. A second revision of the SCFUM resulted in a more inclusive model, with the possibility of a higher-degree of accuracy in the SCI index calculation.

c) *Fine-tuning the existing SCFUM framework.*

The first version of the SCFUM model was based on proprietary ERP systems and professional business experience. Application of the model to actual business data provided the opportunity to calibrate the framework to ensure comprehensive business coverage.

Table 16: Business Specific Entities N/A Exclusion Entity Mapping

Case Ref	Entity No.	Entity Name	SCFUM Initial Mapping	SCFUM Process Node	SCFUM Functional Element
Case G	50002	Document Logging Setup	N/A	Finance	Finance Mgmt
Case G	50004	Pack Type	N/A	Production	Production BOM
Case G	50006	Pack Sub-Type	N/A	Production	Production BOM
Case G	50007	Prepared	N/A	Production	Production BOM
Case G	50008	Crop Type	N/A	Production	Production BOM
Case G	50009	Item Cost Adjustment	N/A	Warehouse	Item Journals
Case G	50010	Master No. Series	N/A	Procurement	Vendors

7.6 Variation Analysis across the SCFUM Revisions

The business cases were iteratively analysed and the model revised through the implementation of the updates described in Section 7.5. The second revision of the SCFUM model (**SCFUMv2.xls)** was found to be more accurate with each new business case applied. The accuracy is shown in Table 15 and Table 18, as the quotient of the **Non-Mapped Entities %** values compared to the first application of the model. A decrease in this variance was evident in each case. The full SCFUMv2 model can be found under Appendix A – SCFUMv2.

A comparison of the two models highlighted the additional process nodes and functional elements, which expanded over the first SCFUM. These covered new business processes within the model, as well as calibrating existing ones for further accuracy. These revisions enabled the model to represent a more accurate business complexity view. A list of these variation updates to the SCFUM is presented in Table 17.

Table 17: SCFUM Process Node and Functional Element Variations

Process Node	**Process Functional Element**	**Description**
Administration	Administration	System administration and setups
	Business Integration	Management and processing of business integration and interfaces
CRM	CRM Management	Contact Relationship Management
Finance	Finance Mgmt	Management of Finance system
	Intercompany	Support for intercompany transactions in a multi-company environment
	Tax Mgmt	Management of Tax returns
	XBRL Mgmt	XBRL Business reporting
HRM	Agency Mgmt	Management of external agency labour
	Benefits	Management of employee benefits
	Disciplinary	Management of employee disciplinary records
	Employee	Management of employee personal records
	Employee portal	Employee reporting and data access through portal
	Expenses	Management of employee expenses
	HR Interfaces	Management of HR system interfaces
	Interview Mgmt	Interview process management and records
	Payroll	Management of employee payroll
	Time & attendance	Time and attendance records
	Workflow	Management of process workflow
Jobs	Jobs	Job / Project Management
Logistics	Bookings	Management of booking records for trips handling requests
	Courier Mgmt	Management of courier records and transactions
	Fleet Mgmt	Management of transport vehicle fleets
	Loading Slots	Management of loading / unloading driver appointment slots
	Logistics Ledger	Audit trail for transport charge transactions processing
	Route Optimisation	Route optimisation for transport planning
	Stock Load positions	Records for stock load positions on trips
	Transport Agents	Management of transport agent records
	Transport Charges	Management of transport related charges
	Transport Contract Mgmt	Management of transport contracts
	Transport Goods	Management of transport goods and products
	Transport Locations	Management of Transportation locations, hubs, depots records

Process Node	Process Functional Element	Description
	Transport Planning	Planning for transportation and trips
	Transport Requests	Management of transportation requests and requirements
	Transport Rules	Management of transport rules for handling requests
	Trips	Management of Trips, Loads, Trucks within transportation
Procurement	Consignment Mgmt	Management of consignment product reporting, traceability and profitability
	Manifest Mgmt	Management of manifest records, import of advanced shipping notices, port deliveries
Manufacturing	Assembly Kit Mgmt	Management and processing of assembly kits
	Label Room Mgmt	Management of label printing process for product, tray-ends, pallets
	Touchscreen Interface	Processing of production transactions through touchscreen interface
QC	Mobile QC	Processing of QC testing through mobile interfaces
	QC Archive	Archive for QC tests and records
	QC Results	Processing and evaluation of QC testing results, stock blocking and updates
	QC Schedule	Management of QC testing plans and schedules
	QC Tests	Management of Quality control and quality assurance testing
Sales	Complaints Mgmt	Management and processing of product related complaints
	EDI Interface	Management and processing of Electronic data interchange interfaces
	Market Trading Mgmt	Management and processing for market trading
	Promotions Mgmt	Management of production promotions and opportunities
Services	Services Mgmt	Service Management
Warehouse	Device Integration	Management and processing of device interfaces and integration
	Mobile Processing	Management and processing of stock through mobile interfaces
	Pallet Mgmt	Management and processing of pallets and pallet tracked product
	Stock Analysis	Analysis and reporting of stock transactions and inventory
	Stock Card	Management of stock records and product details

The revised SCFUM model was then validated by going through a new iteration of the mapping exercise, applying the framework again to the business cases, and re-analysing the subsequent results. The full dataset was processed in filesets **Case#_SCFUMv2map.xls**. The results confirmed all case entities could now be mapped to the SCFUM model.

Table 18: SCFUMv2 Mapping Results

Case Ref	Entities with records	Merged Entities	(N/A) Non-Mapped Entities	(N/A) Non-mapped Entities %	Business Specific Entities
Case A	1,676	876	-	0%	61
Case B	2,119	2,119	-	0%	100
Case C	2,685	623	-	0%	146
Case D	814	544	-	0%	74
Case E	3,143	566	-	0%	75
Case F	2,787	865	-	0%	44
Case G	543	543	-	0%	103
Case H	497	471	-	0%	104
Case I	371	371	-	0%	69

Application of the second revision of the framework to the same business case extract from Table 14 is presented in Table 16. This highlights how an evaluation of this business case now represents zero variances, since both non-mapped entities, including mods, have now been factored-in. Future variances within an SCF framework application would now provide a fairly accurate estimation of the modification requirements that could be required for that organisation.

7.7 Calculating a Baseline SaaS Complexity Index (SCI) for the Unified Model

The key validation step in the research methodology for the SCF framework is its application to actual business case data. Application of the SCM metric to the SCFUM model however provides a baseline SCI index. This could serve as a comparison benchmark to standard business complexity and related deviations, when the framework is applied to business data. Application of the SCFUM model has two important stages within this framework:

1) The derivation of a baseline SCI index for the SCFUM model, to be used within the case study analyses.
2) The derivation of a SCI index for each business case.

The resultant dataset from this analysis would then enable a review of overall business case complexity and the comparative relation of each of these to the baseline SCI index. As detailed earlier in this thesis, the SCM metric has been developed to synthesise a complexity estimate for a business. According to Muketha, et al. (2010) and Fenton, et al. (1997), the process of defining new metrics involves three steps:

i. identify measurement entity,
ii. identify attributes of the entity that are to be measured, and
iii. define new metrics that can be used to measure each attribute.

These steps were followed in the compilation of the SCM metric. As highlighted by Spiteri (2012a)**,** business complexity can be defined as a measure of tangible complexity, which is a composite of business and technological elements determined by these factors:

- number of sub-processes.
- process interactions, interdependencies.
- process context and relationships.
- process environment.

As explained in Section 5.3, this research proposes that there is an inferred singular relation between the complexity of the organisation and the number of interrelated process elements that contribute to its business. This concept was applied in the development of the SCM metric. When this complexity metric was applied to the applied SCFUM model case map, the derived baseline complexity index was equal to **71 (SCI)**, this being the sum of the specific process functional elements within the model, represented as:

$$\sum_{PE=1}^{n} SCFUM\ Process\ Elements$$

These business case datasets, serving as actual representations of complex organisations, therefore proved ideal candidates for prototyping a SaaS transition mode. This having been done through evaluation of the close alignment of their ERP systems to related business processes, factoring concepts described by Kimberling (2008).

Table 19: Case Pivot Data Framework Application

Process Node	Process Element	Sum of No. of Records	Sum of Size (KB)	SC Index
Procurement	Consignment Mgmt	1825	2080	1
	Credit Note	1066	1128	1
	Production Orders	8	32	1
	Purchase Invoices	39365	33656	1
	Purchase Orders	39736	35480	1
	Purchase Prices	310	264	1
	Requisitions	11	408	1
	Stock Card	39	112	1
	Vendors	138600	76400	1
Production	Assembly Kit Mgmt	3	32	1
	Capacity Journals	215	320	1
	Consumption	119	248	1
	Consumption	17	64	1
	Device Integration	4292846	892120	1
	Firm Planned Production Orders	781	2752	1
	Label Room Mgmt	2	192	1
	Machine Centre	8	96	1
	Output	762	808	1
	Planning Worksheets	1297	760	1
	Production BOM	203	584	1
	Production Forecasts	8	224	1
	Production Orders	299	400	1
	Production Routings	64	192	1
	Production Schedule	251	320	1
	Resources	7	80	1
	Shop Calendars	38	96	1
	Touchscreen Interface	347	392	1
	Work Shifts	4	32	1

To this effect, recalculation of the baseline complexity for the second SCFUM revision results in a baseline complexity index of **124 (SCI)**, this being calculated as:

$$SCI_{(SCFUMv2)} = \sum_{PE=1}^{n} SCFUMv2\ Process\ Elements$$

Where new business functional elements have been included, new processes nodes were in some cases also required. The net result of this exercise was the increase of new processes and functional elements over the original SCFUM model, listed in Table 17, with the net variance between SCFUM and SCFUMv2 being **53 (SCI).** It should be noted that the revised SCFUMv2 also consolidated three Process Elements from the original SCFUM model, into one.

7.8 Summary

In this chapter, the framework was applied to business case study datasets, as part of the Step 5 in the research methodology. A step-by step worked example was described, using same extracts from CASE G throughout. This showed the selection and preparation of the raw datasets, ready for application. This process then enabled the validation and review of the SCFUM model for Step 6 of the methodology, ensuring comprehensive coverage for business specific and industry specific entities. The variances across the framework applications were then investigated, and the baseline complexity index re-calculated for the revised unified model.

The SCM metric was subsequently applied directly to SCFUM model, enabling the calculation of a baseline complexity index, to be used over the next chapters as part of the analysis discussion around the business case datasets.

CHAPTER 8

CASE SCI ANALYSES AND DISCUSSION

8.1 Introduction

Expanding on the work done in the previous chapter, the results generated from the application of the SCF framework were reviewed and discussed. As part of Step 7 of the research methodology, the SCI indices calculated for each business case were applied in a comparative analysis, deriving additional statistics on the data. The discussion around the output results presents ways on which this complexity data can be interpreted within an organisation, and its importance within a SaaS solution compliance review. This was achieved within a factual commercial context.

8.2 SCI Business Case Analyses

The SCI index for each of the business cases was calculated using several pivot tables, derived from the analyses referenced in Chapter 7. These analyses collate the results from the mapping of the Process Nodes and Process Elements constituting the SCFUM model. This provided a detailed overview of the process structure for each business case. This process was undertaken for all nine case datasets, together with a comparison against AX and NAV, as exemplars of proprietary off-the-shelf ERP systems. Extracts from **Case#_SCFUMv2map.xls** and **SCFUMv2 mapping stats.xls** filesets show representations of these pivot table analyses in Figure 29, Table 19, Table 20 and Table 21.

The pivot tables in **Case#_SCFUMv2map.xls** were derived, consisting of four main information fields, as shown in Table 20:

1. **Process Nodes** – the mapped process nodes applicable to that dataset.
2. **Process Elements** – the mapped process functional elements applicable to that dataset.

3. **Count of Process Elements** – the count of the total number of process functional elements within that dataset.
4. **SCI Index per Process Element** – the SCI index applied to each individual process functional element.

Table 20: Pivot SCI Analysis Extract (Case#_SCFUMv2map.xls)

Process Node	Process Element	Count of Process Element	SCI Index
⊟ **Sales / Procurement**	Credit Notes	5	1
	Logistics Ledger	1	1
	Payment Journals	6	1
	Payment Methods	34	1
	Payment Terms	5	1
	Stock Card	3	1
	Tax Mgmt	1	1
	Transport Planning	3	1
	Vendors	1	1
	XBRL Mgmt	2	1
⊟ **Services**	Budgets	1	1
	Chart of Accounts	2	1
	Customers	1	1
	General Ledger Registers	2	1
	Production Schedule	12	1
	Resources	4	1
	Sales Prices	23	1
	Service Mgmt	1	1
	Services Mgmt	112	1
	Work Shifts	2	1
⊟ **Warehouse**	Assembly Kit Mgmt	31	1
	Bin Contents	4	1
	Business Integration	1	1
	Device Integration	3	1
	Inventory	76	1
	Item Journals	14	1
	Market Trading Mgmt	4	1
	Pallet Mgmt	3	1
	Purchase Prices	7	1

Through these datasets, each case study was analysed individually via its own pivot table. These were subsequently consolidated for further processing within the **SCFUMv2_mapping_stats.xls** file, enabling further detailed analysis comprising the following data fields:

1. **SCF Process Nodes**: the mapped process nodes applicable to the datasets.
2. **Case Ref**: Reference to each of the anonymised business case studies.

3. **Total No. of Records / Month** – total number of records, per Process Node, averaged out for a month for each business case.
4. **Total Average Record Size (B)** – this represents the total average record size in bytes, for that case reference and process node.
5. **Total Sum of Size (KB) / Month** – this is the total record size in KB, averaged for a month of operations, for that business case, and specific process node.
6. **Total Count of SCF Process** Element – this field holds the total count of mapped process functional elements, linked to that SCF Process Node.

Table 21: Pivot SCF Process Node Analysis Extract (SCFUMv2 mapping stats.xls)

SCF Process Nodes	Case Ref	Total No. of Records / Month	Total Average of Record Size (B)	Total Sum of Size (KB) / Month	Total Count of SCF Process Element
⊟Production	Case A	4,890	9,904	4,719	106
	Case B	2,242	13,316	1,537	53
	Case C	12,112	12,531	9,760	46
	Case D	113	9,619	182	74
	Case E	7,429	10,449	1,310,864	73
	Case F	14,216	11,334	10,253	96
	Case G	326,938	10,700	148,054	96
	Case H	41,578	13,833	19,669	77
	Case I	2,312	10,853	2,015	25
Production Total		**411,830**	**11,331**	**1,507,053**	**646**
⊟QC	Case A	1,761	4,028	1,025	21
	Case B	11,260	2,868	3,798	15
	Case C	22	14,459	273	18
	Case D	1	13,961	9	13
	Case E	39,753	3,656	174,226	18
	Case F	115	7,305	165	14
	Case G	373,012	1,178	168,943	19
	Case H	42,481	1,063	19,416	18
	Case I	0	12,288	2	2
QC Total		**468,405**	**6,742**	**367,856**	**138**
⊟Sales	Case A	20,764	9,482	12,491	104
	Case B	432,705	12,302	224,782	89
	Case C	413,381	8,182	243,101	91
	Case D	207,792	6,175	179,925	71
	Case E	27,123	12,378	1,522,511	93
	Case F	36,422	21,863	44,032	98
	Case G	178,460	3,692	105,904	76
	Case H	32,086	6,013	23,264	63
	Case I	18,534	6,604	16,689	52
Sales Total		**1,367,266**	**10,200**	**2,372,698**	**737**
⊟Sales / Procurement	Case A	3	12,971	41	4
	Case B	19	14,557	15	4
	Case C	15	17,207	231	4
	Case D	0	17,164	3	3
	Case E	15	20,480	205,824	4
	Case F	27	7,798	169	4
	Case G	2	19,590	8	4
	Case H	1	21,163	4	3

Through this application of the SCF framework and its SCM metric, the resultant data derived from this pivot table (Table 21), enabled a comprehensive view on the volume of transactions each of the organisations in the case studies process in a month. In this manner, several key SaaS service estimates can be calculated, providing an estimated operating cost for that business, accurate to the process functional element level of the framework.

8.2.1 Calculation of the SaaS Complexity Index for the Business Cases

Following the application of the revised SCFUMv2 model to the business cases, it became possible to derive an accurate SCI index for each of the case datasets. The calculation criteria used were as follows:

$$SCI_{(Case\ N)} = \left(\sum_{PE=1}^{n} SCFUMv2\ Process\ Elements + \sum_{MPE=1}^{n} Mods\ Process\ Elements \right), \text{"}No. of\ Records\text{"} \neq \Phi$$

Where ∑ *SCFUMv2 Process Elements* (PE) is the summation of distinct process functional elements within the case datasets, and ∑ *Mods Process Elements* (MPE) is the summation of distinct business specific elements (or modifications on process functions). This applied only to process elements where the "No. of Records" is not null.

Table 22: Business Case SCI Indices

CASE SUMMARY	SC Index
SCFUMv2 Baseline	124
Case A	112
Case B	98
Case C	93
Case D	92
Case E	96
Case F	119

CASE SUMMARY	SC Index
Case G	97
Case H	91
Case I	69
Average Index	99

Table 22 summarises this analysis for each business case SCI, whilst Figure 30 represents a graphical comparison of these indices within the whole research dataset.

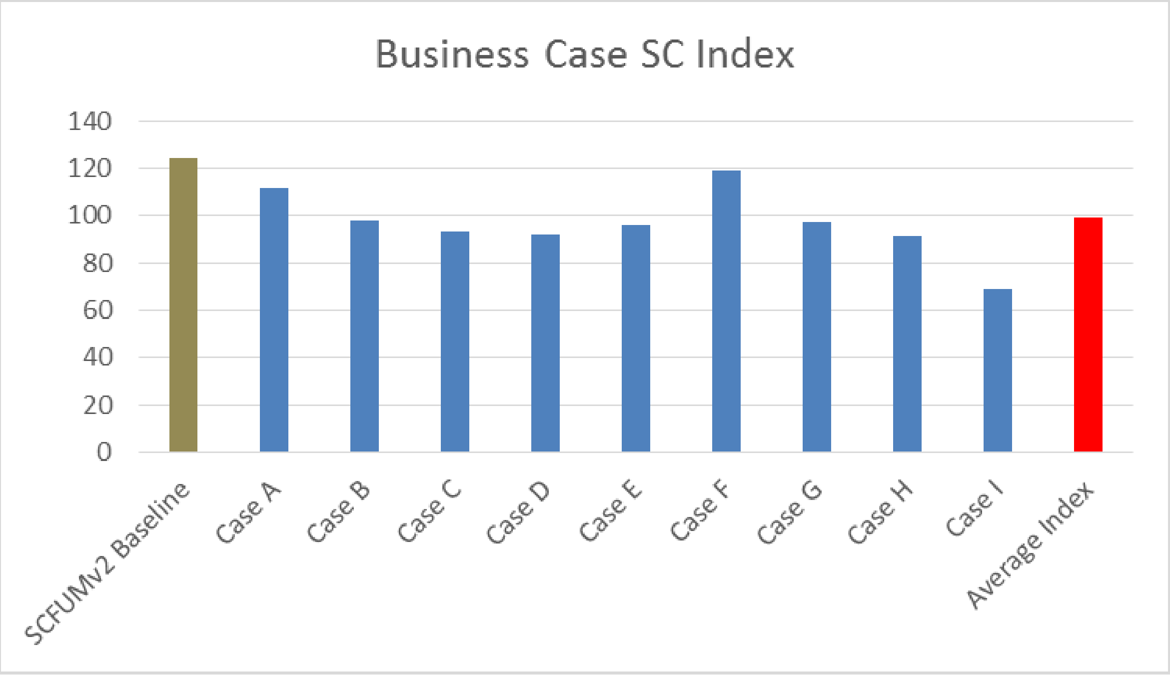


Figure 30: SCI against Business Case - Complexity Results

Analyses of these results yields strategic information, that would make it invaluable in supporting key commercial decisions when investing in SaaS hosted services. The following are some examples of possible information that could be inferred by an organisation:

i. With an SCI of 119, business CASE F could be described as more complex than business CASE I, and its SaaS transition is therefore likely to involve a higher risk, cost and time.

ii. Since the business cases have been extracted from the same industry, an average SCI of 99 suggests the optimum investment level for this industry. A potential service provider could make use of this SCI to achieve a solution design that balances the business complexity needs for majority of the market, in relation to the investment made.

iii. Calculating the difference between the SCI of business CASE A and an appropriate solution covering the average complexity of 99 SCI. This would provide a metric of the level of complexity that is therefore not defined by this proprietary solution. Thus, it enables forecast derivations for level of customisation needs, or business process re-engineering requirements.

This complexity quantification could be factored directly in the calculation of forecasts for costs and time-frames within a possible SaaS solution transition process. These concepts are further discussed in Section 8.2.4 and Chapter 9.

The SCF framework data results could hereafter be used to calculate more accurately service bandwidth, and cloud capability requirements for each business. This would in real terms translate to the possibility of deriving forecasts for monthly operating expenses for the organisation, as well as VM bandwidth required to resource an ERP SaaS solution. During the solution decision process, this would support strategic decisions in modelling that organisation's business processes to the cloud hosted service model.

The BI capabilities of pivots tables enabled a comparison of the SCI complexity for each business case (Section 8.2.1), grouped at the SCF Process Node level, as shown in Table 23.

Table 23: Total Case SCI Index / SCF Process Node

Total Case SCI Index per SCF Process Node

Sum of SCI	SCF Process Nodes													
Case Ref	**Administration**	**CRM**	**Finance**	**HR**	**Jobs**	**Logistics**	**Procurement**	**Production**	**QC**	**Sales**	**Sales / Procurement**	**Services**	**Warehouse**	**Total SCI Index**
Case A	1	3	18	11	1	14	9	17	4	14	3	1	15	111
Case B	2	2	15	4	1	13	9	14	4	16	3	1	14	98
Case C	1	2	15	3	1	11	9	14	4	14	3	1	14	92
Case D	2	1	14	4	1	10	9	18	2	11	3	1	15	91
Case E	1	2	13	2	1	12	9	17	3	16	3	1	15	95
Case F	1	3	18	12	1	14	9	19	3	18	3	1	16	118
Case G	2	2	13	5	1	12	8	18	2	15	3	1	14	96
Case H	1	2	13	4	1	11	9	17	2	14	2	1	14	91
Case I	1	2	13	4	1	7	8	12	2	11	2	1	10	74
Total SCI Index	**12**	**19**	**132**	**49**	**9**	**104**	**79**	**146**	**26**	**129**	**25**	**9**	**127**	**866**

To provide some examples, these SCI complexity comparison results demonstrate that:

- CASE I, with a total SCI of 74, shows as less complex than CASE F which has a total SCI of 118.
- CASE A returned a higher complexity within the Human Resource Management (HR) node, of 11 SCI, than the same for CASE B which resulted in 4 SCI.

The SCI per Process Node in Table 23 derives the process complexity for each distinct node within a business case dataset. Table 24 shows a calculation of the total count of the SCF process elements per node, as a percentage of the sum of process elements for that case reference. This quantifies as a percentage the complexity of a specific process node as a ratio of the overall SCI complexity. Within a business case, analysis of these results returns a value for the functional complexity and functional depth requirements for each specific SCF Process Node.

A solution for that business would therefore need to support both the main processes, as well as the functional complexity required at the process element level. The formula applied here is:

$$SCF\ Process\ Element\ \%_{(Process\ Node)} = \frac{\sum Process\ Element_{(Process\ Node)}}{\sum Process\ Element_{(Case)}} \times 100$$

Table 24: Process Element % Derivation per Node and Case

Case Ref	Administration	CRM	Finance	HR	Jobs	Logistics	Procurement	Production	QC	Sales	Sales / Procuremen	Services	Warehouse
Case A	3%	5%	23%	4%	2%	6%	7%	12%	2%	12%	0%	5%	17%
Case B	7%	1%	27%	1%	0%	9%	9%	8%	2%	14%	1%	0%	20%
Case C	6%	3%	29%	1%	0%	5%	10%	7%	3%	15%	1%	0%	21%
Case D	4%	0%	25%	2%	0%	5%	13%	14%	2%	13%	1%	0%	21%
Case E	5%	2%	21%	1%	1%	6%	9%	13%	3%	16%	1%	2%	21%
Case F	4%	6%	23%	5%	2%	6%	7%	11%	2%	11%	0%	5%	18%
Case G	4%	2%	22%	2%	0%	6%	7%	18%	3%	14%	1%	0%	21%
Case H	4%	1%	21%	3%	0%	5%	7%	16%	4%	13%	1%	0%	24%
Case I	6%	3%	32%	2%	1%	4%	11%	7%	1%	14%	1%	0%	20%

Table 25 shows a worked example for the "SCF Process Element %" calculation for the CRM process node in respect to the nine case studies. As an example, for CASE G, it can be inferred that a cloud solution with a stronger finance element is more relevant than one with a services focus.

Table 25: SCF Process Element % Calculation Example

Process Node	Case Ref	Total Count of SCF Process Element	Total SCF PE Case Count	Percentage SCF Process Element
CRM	Case A	47	876	5%
CRM	Case B	8	635	6%
CRM	Case C	17	623	3%
CRM	Case D	2	544	6%
CRM	Case E	11	566	7%
CRM	Case F	50	865	2%
CRM	Case G	13	543	4%
CRM	Case H	3	471	5%
CRM	Case I	12	371	13%

An innovation in the SCF framework is therefore that it offers the opportunity to compare and contrast the complexity for all the business case studies, using the SCI complexity indices for each of these. Figure 31 shows a representation of this comparison in bar chart format. Note how business CASE F stands out as being the most complex within this study at 118 SCI, whilst in contrast CASE I show as the least complex business, with a 74 SCI.

This cross-section of business case complexity allows the calculation of an average complexity index for the industry that these businesses operate in, that is, food processing. This was calculated as having a complexity of 96 SCI.

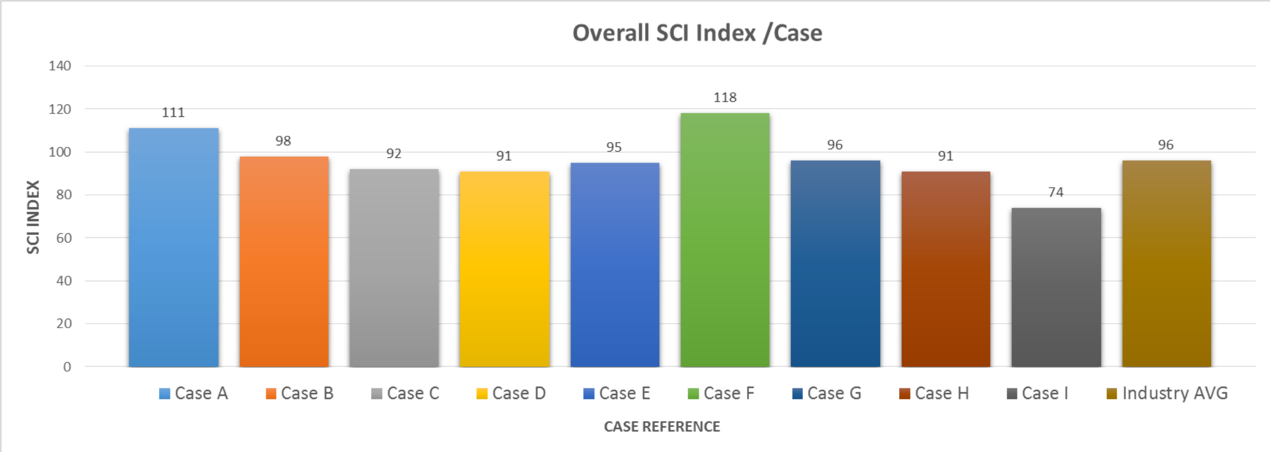


Figure 31: Bar Chart for Total SCI Index per Case

The SCF framework here provides a means for a more detailed cross-section analysis of the average SCI complexity for this industry. As shown in Table 26, an SCI is derived for each process node within the framework, indicating the following:

- **Min SCI / Node**: This is the minimum complexity identified across the business case studies for that process node.
- **Max SCI / Node:** This is the maximum complexity identified across the business case studies for that process node.
- **Average SCI / Node**: This is the median complexity identified across the business case studies for that process node.
- **Total Average SCI:** This sum-total of the Average SCI per Process Node column. Since the business cases are samples from a common industry, then this value exposes the **Industry Average SCI Index**.

Table 26: Average Industry SCI Index per SCF Process Node

SCF Process Node	**Min SCI /Node**	**Max SCI /Node**	**Average SCI /Node**
Administration	1	2	1
CRM	1	3	2
Finance	13	18	15
HR	2	12	5

SCF Process Node	**Min SCI /Node**	**Max SCI /Node**	**Average SCI /Node**
Jobs	1	1	1
Logistics	7	14	12
Procurement	8	9	9
Production	12	19	16
QC	2	4	3
Sales	11	18	14
Sales / Procurement	2	3	3
Services	1	1	1
Warehouse	10	16	14
SCI TOTALS	**71**	**120**	**96**

This data could be used by potential service providers in evaluating how their bespoke solution would comply with the ERP complexity requirements of a specific industry. This metric could additionally assist and inform how to tailor an ERP solution to meet the functional requirements of a specific process. For example, from this analysis, a solution for a food processing industry should cover an overall complexity requirement of 96 SCI, with a functional focus on Finance, Production and Logistics, as opposed to CRM and Services, for which results indicate are less relevant.

Evaluating these SCI results graphically as an overlay of Industry, Average and Min/Max provides a useful visual cue of how the various SCF process nodes compare across the whole case datasets. This view is represented in Figure 32, for this common case study industry.

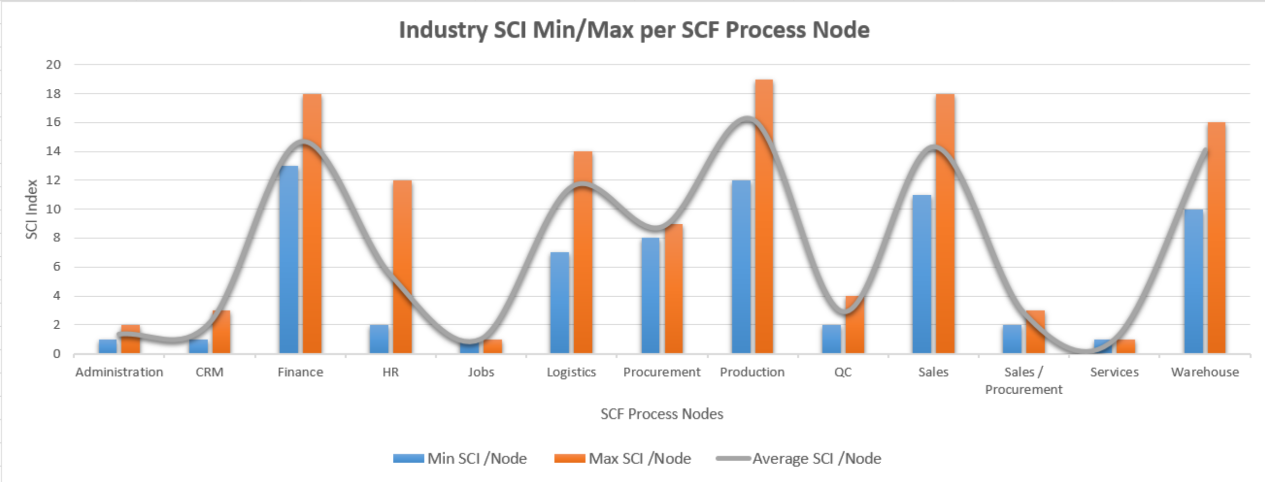


Figure 32: Industry Min/Max SCI per Process Node Chart

8.2.2 Case Data SCF Process Element Comparison

As described in the previous section and represented in Table 24, analysis of the SCF process elements evidenced the functional depth required for each process node within a business case. The functional depth provides another business dimension to the information yielded through the calculation of the SCI complexity. The figures below show a representation of how the various business cases compare for overall process element sub-divisions across each organisation. Pie charts have been used to demonstrate the distribution of process elements for this purpose.

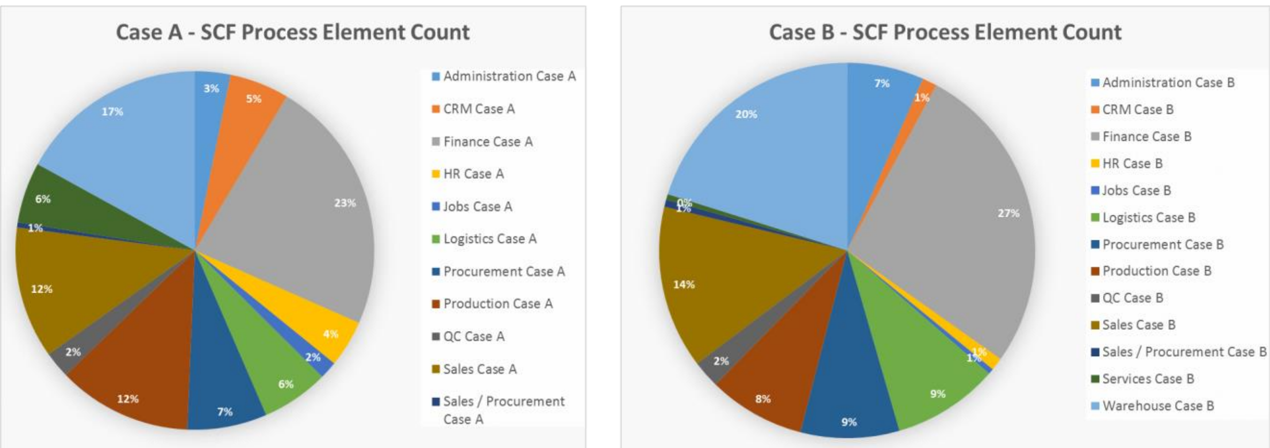


Figure 33: SCF Process Element Count Chart A - B

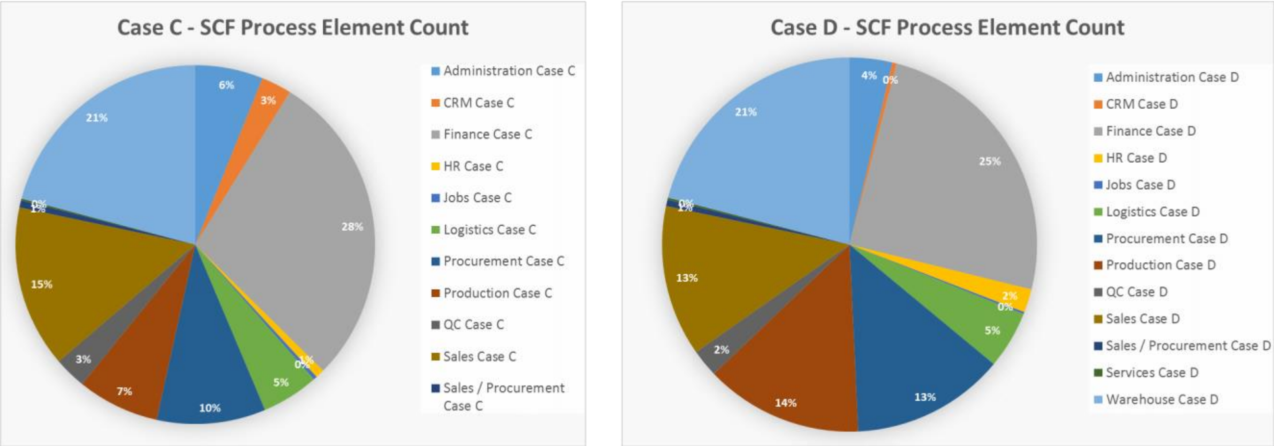


Figure 34: SCF Process Element Count Chart C - D

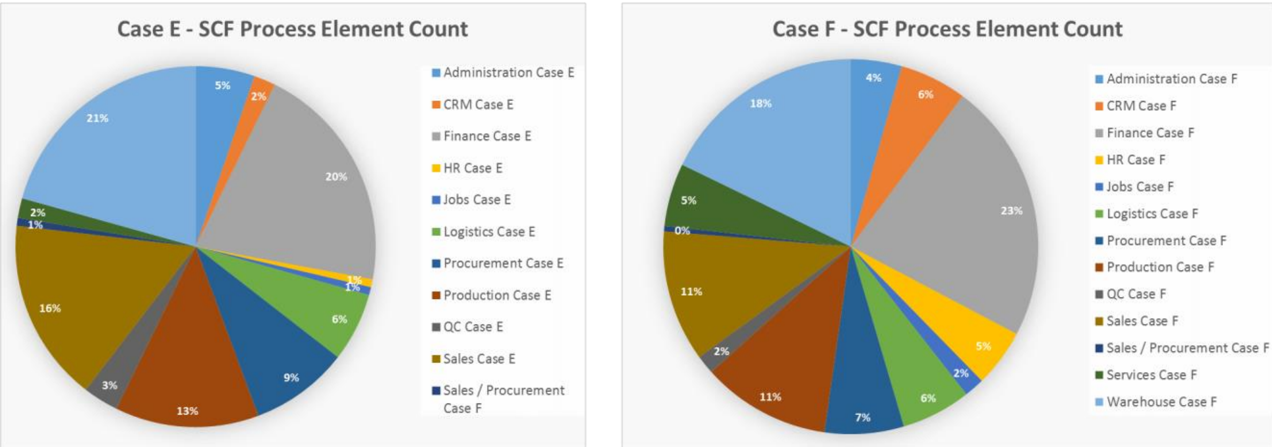


Figure 35: SCF Process Element Count Chart E - F

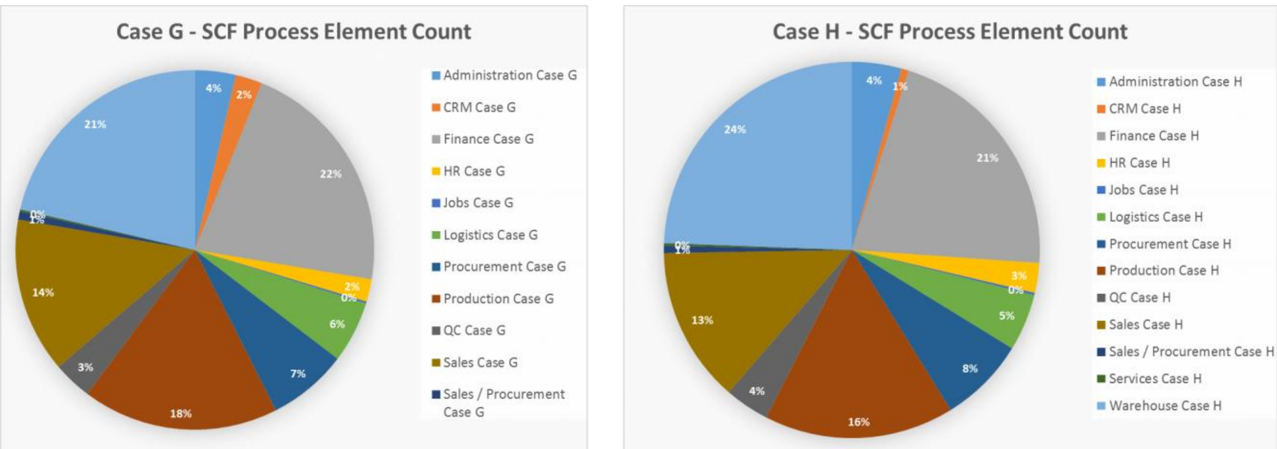


Figure 36: SCF Process Element Count Chart G - H

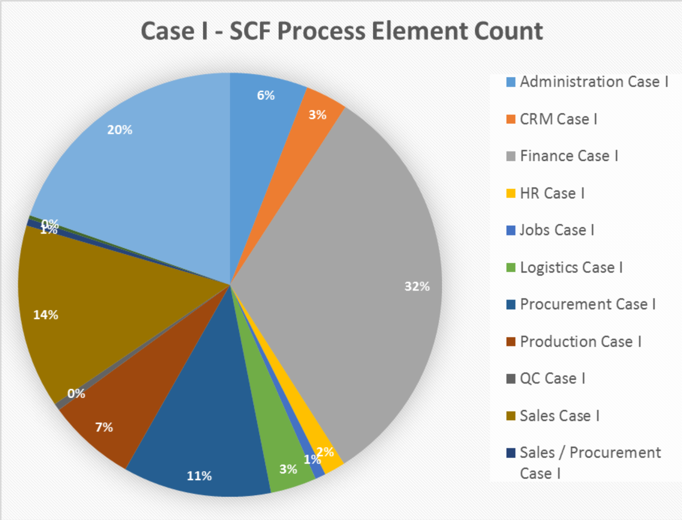


Figure 37: SCF Process Element Count Chart I

These comparative representations serve to highlight how even though organisations could have similar complexity measures within their overall process, the functional requirements within each could vary substantially between them. The SCF framework allows for this differentiation, and enables the derivation of a view for both. As an example, whereas both CASE D and CASE H have a comparative complexity of 91 SCI; CASE D has more elaborate finance requirements, and simpler QC requirements than CASE H. In real terms, this could result in differing solutions being found appropriate for each business.

8.2.3 ERP Solutions Evaluation through SCF

In traditional implementation models, ERP systems and solution providers are commonly evaluated by a business through a request for proposal (RFP) process. An example of a functional requirements template is presented in Appendix B.

The results from this RFP process provides an overview of how solutions could meet requirements, as well as deriving a comparable indication of cost and risk. This information then forms the basis on which a strategic decision for vendor and solution selection is made. This approach poses some limitations however, in that the process is subjective in nature and relies on the vendor understanding requirements and responding with a suggested level of compliance of the existing solution. Additionally, within a SaaS context there is an expectation that the service consumer does most of the service evaluation themselves prior to subscription. These changes in the implementation process are further detailed in Chapter 9.

The SCF framework mitigates these limitations, and aims to facilitate the evaluation process for the service consumer. As a representative example of this, the framework was applied to off the shelf ERP solutions, namely Microsoft Dynamics NAV and Microsoft Dynamics AX, processed in files **NAV_Vanilla_SCFMap.xls** and **AX2012R3_SCFMap.xls** respectively. This was done by extracting the functions covered by each solution and the business modules that these operate in. The results returned from these analyses provided insight into the level of SCF Process Node and SCF Process Element complexity covered by each of these solutions. The dataset for this part of the analysis was processed and stored in file **ERP_SCI_Analysis.xls,** with tabular extracts from this represented later in these sections. Whilst the versions of the ERP systems described in Section 4.4 were on-premises solutions, the same concepts would apply to cloud hosted solutions.

The results obtained through SCF were the derivation of the SCI complexity index for the two ERP systems. The lower cost Microsoft NAV returned an overall complexity of **93 SCI** and the higher cost Microsoft AX a complexity of **227 SCI**. This immediately highlights how AX could be more appropriate to more complex businesses, since standard functionality would meet more of the business requirements. Additionally, applying the same AX solution to a less complex business could prove an unnecessary overhead in cost and complexity with the implementation.

Table 27 details an example of how these results could be used by IT Managers for a potential business procurement in comparing the various solutions applicable for that business. For this example, scenario business CASE F was used since its SCI is between the SCI for the two ERP systems. From this comparison, one could infer a multitude of strategic information for possible implementations. Some such examples could be as follows:

- CASE F has an overall complexity of 118 SCI, then NAV is likely to require customisation to meet this business's requirements, whilst AX is likely to be over-specified for these same enterprise requirements.
- The finance process coverage of NAV meets the requirements of CASE F, yet production would require extensive customisations. This can be estimated as an addition of 16% to project costs, calculated as: (((19 SCI – 13 SCI) * (NAV man days per SCI)) / (NAV Days)) * 100.

- AX meets the Quality Control (QC) requirements of CASE F, with NAV offering less functionality in this area, demonstrated through Table 27.

Table 27: Example Case SCI Index Comparison to ERP

Case Ref	Case F	NAV SCI	AX SCI
CRM	3	3	3
Finance	18	18	30
HR	12	12	21
Jobs	1	1	41
Logistics	14	1	9
Procurement	9	10	10
Production	19	13	27
QC	3	0	3
Sales	18	16	42
Sales / Procurement	3	3	10
Services	1	1	10
Warehouse	16	13	18
Total SCI Index	**118**	**93**	**227**

As detailed in the previous section, the application of the SCF also enables a view of the functional depth of the various SCF process nodes across the overall ERP solution. These results are represented in Table 28 and
Table 29 which detail the "SCF Process Element %" coverage for each system.

Table 28: NAV ERP SCF Process Element % Coverage

NAV SCI		
SCF Process Nodes	**Count of SCF Process Element**	**Percentage SCF Process Element**
Administration	60	6%
CRM	67	7%
Finance	334	34%
HR	63	6%

Jobs	23	2%
Logistics	3	0%
Procurement	44	5%
Production	92	9%
Sales	76	8%
Sales / Procurement	5	1%
Services	67	7%
Warehouse	142	15%
Grand Total	**976**	**100%**

Table 29: AX ERP SCF Process Element % Coverage

AX SCI		
SCF Process Node	**Count of SCF Process Element**	**Percentage SCF Process Element**
Administration	727	9%
CRM	63	1%
Finance	1910	24%
HRM	846	10%
Jobs	351	4%
Logistics	202	2%
Procurement	772	10%
Production	454	6%
QC	8	0%
Sales	1654	20%
Sales / Procurement	61	1%
Services	160	2%
Warehouse	889	11%
Grand Total	**8097**	**100%**

The data in these tables provides important comparative metrics for a business within the system selection process. An example of such a functional depth comparison is detailed in Table 30, which uses results from business CASE F (Figure 35). Some examples of how this data could be interpreted are:

- Depth of finance functions within NAV is more comprehensive than AX, even though AX natively provides wider process coverage (as identified in Table 27).
- The production functional requirements for CASE F show as more elaborate than any of the two systems could offer. This implies that some customisation might still be required within this SCF Process Node, whichever ERP system is selected.

Table 30: Example Case SCI Element % Comparison to ERP

Case Ref	Case F	NAV SCI	AX SCI
Administration	4%	6%	9%
CRM	6%	7%	1%
Finance	23%	34%	24%
HR	5%	6%	10%
Jobs	2%	2%	4%
Logistics	6%	0%	2%
Procurement	7%	5%	10%
Production	11%	9%	6%
QC	2%	0%	0%
Sales	11%	8%	20%
Sales / Procurement	0%	1%	1%
Services	5%	7%	2%
Warehouse	18%	15%	11%
Total	**100%**	**100%**	**100%**

An easier way to visualise the complexity comparisons between the two ERP systems, for both process nodes, and process functional element coverage is representing these using bar charts. Figure 38 and Figure 39 highlight how AX covers more process complexity requirements across the various SCF process nodes, however NAV still provides more functional coverage in some process nodes, such as for Finance and Warehouse.

An example of what this could translate to in-reality would be that both ERP systems support a Sales Order creation process (SCI Process Node complexity), yet only one system supports a Price Discount functionality within that process, whilst the other does not (Process Element % coverage).

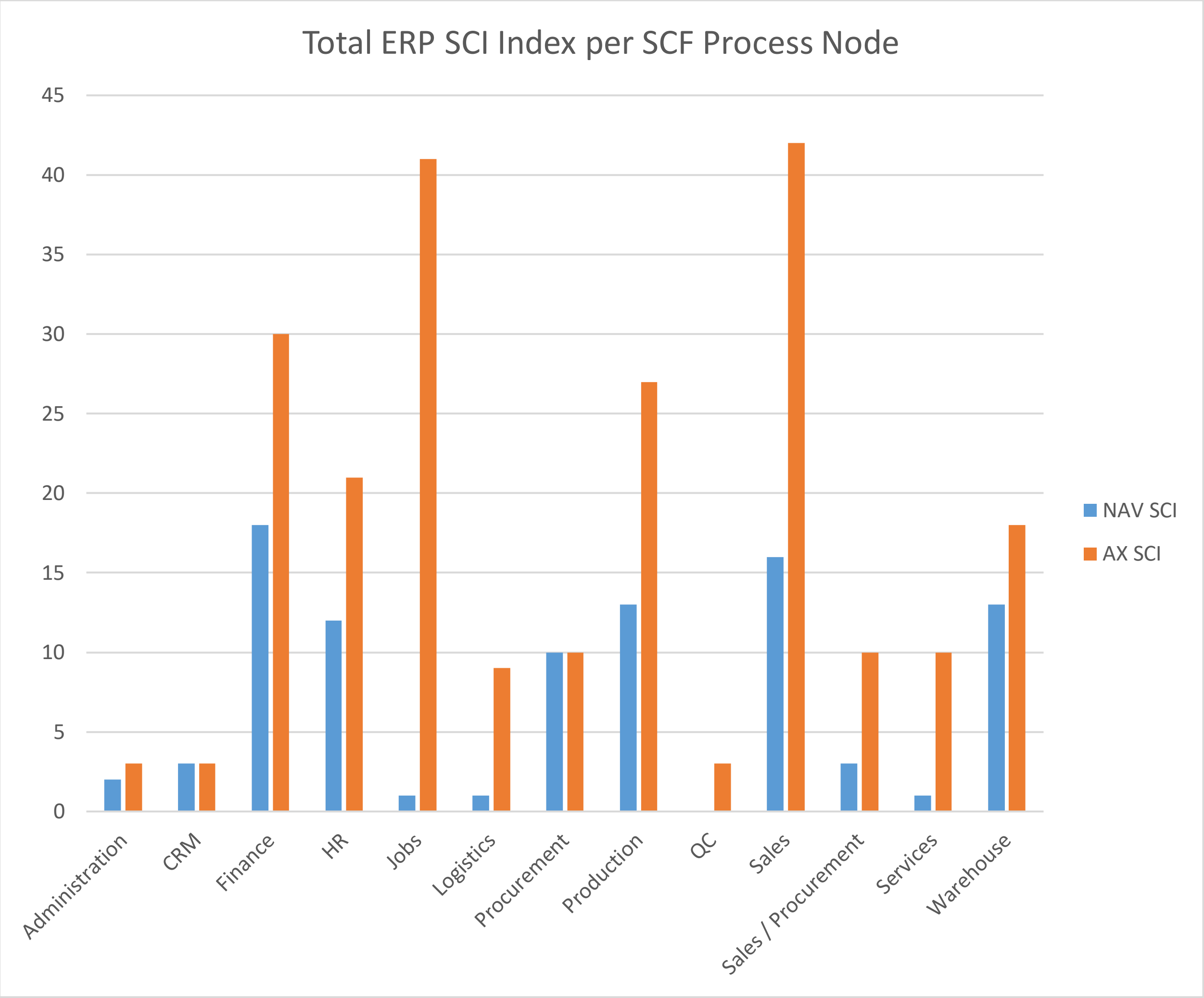


Figure 38: NAV vs AX: ERP SCI Index per SCF Node Comparison Chart

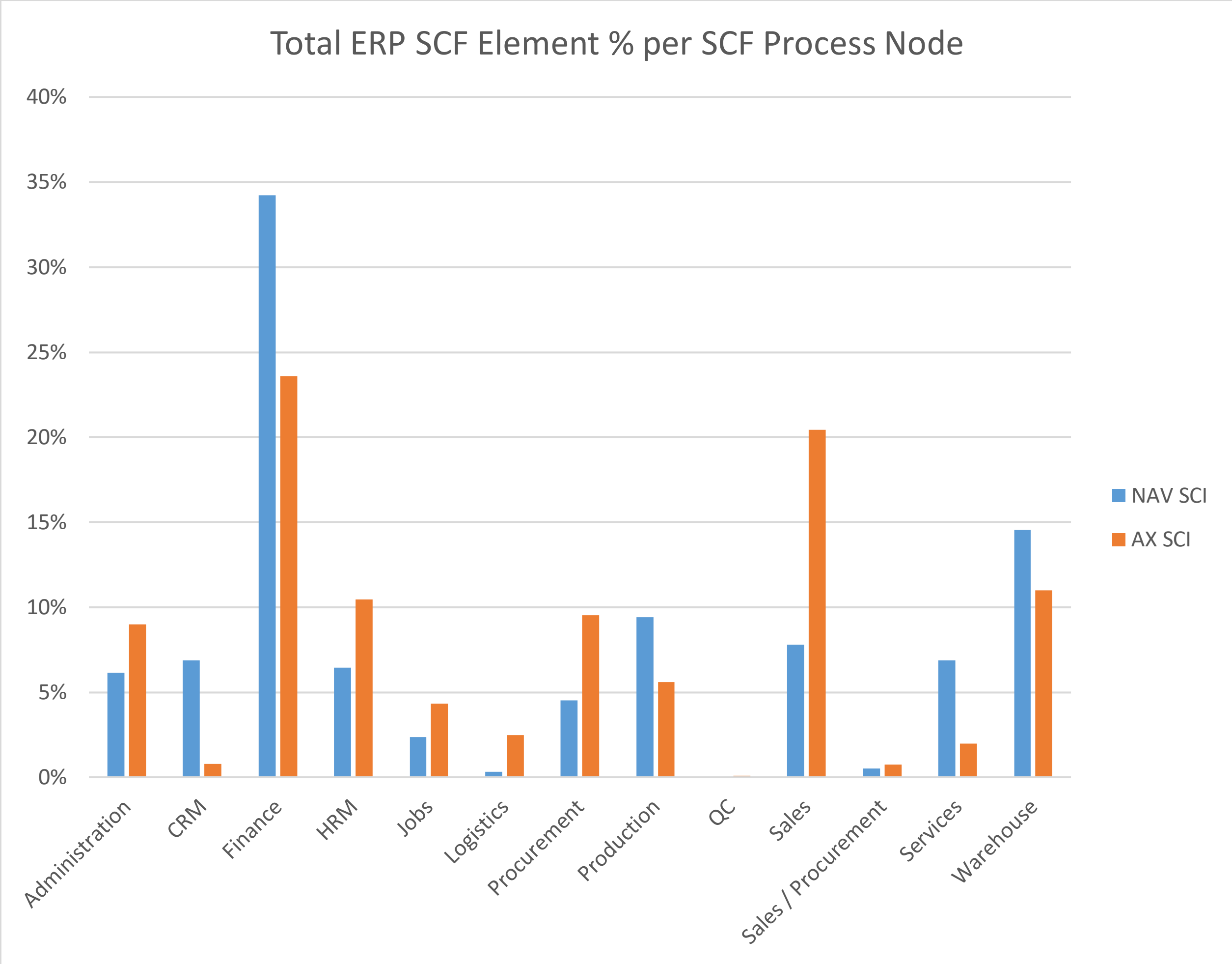


Figure 39: NAV vs AX: ERP SCI Element % per SCF Node Comparison Chart

8.2.4 Example Strategic Implementation Calculations

Once the SCF framework has derived the basic complexity and functional element results for a business or system, there exist a myriad of other possible applications for this data. One example would be a scientific calculation of the possible duration a solution would take to implement and deploy for a business, as well as estimating the related resource requirements. Table 31 identifies the average implementation duration for each of these ERP systems, and derives the number of days required to implement as a unit of SCI complexity.

Table 31: SCI Application in Time-to-Deploy Calculation

Time to deploy	Days	SCI	Work-Days / SCI	Team Size	Man-Days / SCI
NAV (2 years)	730	93	7.849	2.5	19.624
AX (3 years)	1095	227	4.824	5	24.119

- **Time to deploy** – This is the technology analysts' estimate for the maximum average system implementation duration per ERP system, described as work duration.
- **Days –** The time to deploy converted to days. Note: for purposes of this example holidays are not factored in. These would result in a deduction of actual work days.
- **SCI –** The complexity index calculated through the SCF framework per system.
- **Work-days / SCI –** Derived number of work days required per unit of SCI complexity. This is a calculation of elapsed time, independent of number of resources on the project.
- **Team Size** – common team size requirement for implementation projects per project. This is estimated through professional experience.
- **Man-days / SCI** – the man-days required for implementation, per unit of SCI complexity for each ERP system. Through this value, the estimated duration of an implementation is calculated as a total for all time spent on the project by every resource. This is a key calculation, as enables total budgeted cost expectations in support of the project plan.

The general resource requirements for the team structures responsible in implementing each system is highlighted in Table 32. These are common setups, informed through the author's professional experience within actual ERP solution implementations.

Table 32: Common ERP System Implementation Teams

AX Teams	NAV Teams
Lead Consultant / Solution Architect	Lead Consultant / Solution Architect
Trade and Logistics Consultant	Functional Consultant
Finance and Inventory Consultant	Technical Consultant
Production / Manufacturing	
Technical Consultant	

8.3 Summary

Following the case study analyses, this chapter validated that the SCF framework can be applied with a high-level of accuracy to actual business scenarios. Accuracy was measured as a basis of unmapped elements within the application process. Representing Step 7 in the research methodology, the SCF framework was further validated in its application to nine business case studies, obtaining comprehensive quantitative results. Differences encountered in the mapping of the SCFUM model were found to serve as indicators of the levels of business specific characteristics within an organisation. These results would derive a forecast for the custom functionality required to provide an ERP solution which is compliant with that business.

It has been noted, that whilst the case datasets from this research, and subsequent framework mapping process was carried out in relation to traditional ERP system entities, this same process could be similarly applied directly to physical business processes within an organisation. In this case however, the business would be expected to carry out an initial analysis of its internal processes to identify these and structure them in a manner to facilitate the application of this framework. Such an exercise could be carried out within an RFP process, as per the one shown in Appendix B.

Thus, the research demonstrates that a business which has their processes already represented within an existing IT system would have an easier analysis process, and therefore would facilitate any possible SaaS system transition.

CHAPTER 9

Defining a Cloud Transition Strategy for the Service Provider

9.1 Introduction

The objective of this chapter is to present the practical contextual use of the framework within cloud transition strategies. An outline of the technical options to specify a cloud deliverable solution is described, proposing possible strategic alternatives from the service provider's perspective. This has been primarily based on the author's practical experience and adaptations from comparable sciences, applied to business complexity.

A key motivation for traditional service providers for having a cloud hosted solution option is to identify new sustainable revenue streams that add value to their on-premises ERP offerings. This increases their competitiveness in an aggressive industry, and enables them to grow their customer base. Figure 40 highlights the different business profiles for service providers, between Cloud and legacy on-premises solutions.

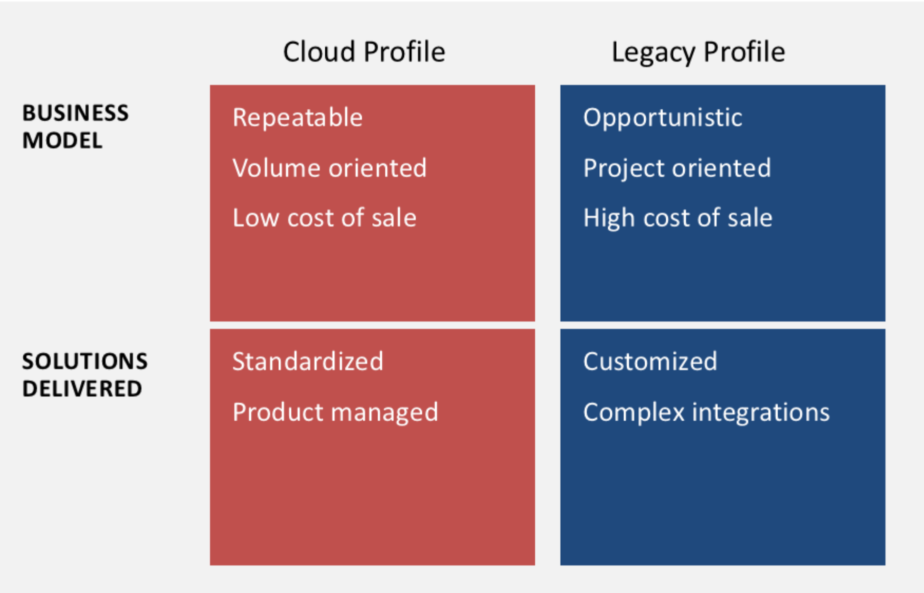


Figure 40: Cloud vs Legacy Profiles (adapted from Microsoft Resources, 2015)

Over the next sections, several cloud strategies are analysed and condensed within a final strategic proposal. The author here considers the wider business and commercial elements to the strategy, required to support the technical demands to ensure a successful technological solution. Supporting the comparative analysis of these strategic options, the author's practical experience as well as the SCI business complexity results derived in Chapter 8 are made use of for estimations to differentiate these options. It should be noted that the SCI complexity range applied here is dependent on the business cases and their respective industry, summarised in Table 26. Application of these strategies against other industries would likely yield a different SCI value range.

9.2 Business Setup Considerations

By 2020 all business would have an element of Cloud within their IT systems (Gartner, 2016). Service providers are taking note of this shift, and legacy service providers have started creating strategies for providing their services in the Cloud. This research indicated that strategies should factor-in the following considerations for a quicker transition:

- Some service providers would be transitioning to a cloud service model, by converting existing on-premises solutions to support SaaS. Precautions should be taken to ensure that this shift in business model does not impact negatively overall profitability. This risk is due to the deferred revenue and profit generation from the cloud subscription model, when compared with on-premises model, which has up-front license payments.
- Cloud service providers would likely need to partner with one of the main players for other cloud services, such as Microsoft Azure, Amazon EC2, or others, creating a level of dependency.
- Solution pricing, commonly in the form of monthly subscription costs per user, as well as initial setup fees, need to be well defined on the onset.
- Consideration should be given to bundled offerings and wrap-around professional services, such as training, provisioning, data migration, configuration, mail setup, SharePoint development and helpdesk, amongst others. These can then all be billed up-front for a fixed price, providing an additional revenue stream.

- A cloud strategy for a service provider would be a key business direction decision, as this could result in far-reaching changes to the business structure and working model of its operations. As such therefore requires involvement of high-level management each step of the way.

9.2.1 Customer Cloud Service Procurement Process

The customer decision journey to procure a SaaS based solution is different from that pursued when purchasing an on-premises ERP solution. The process is intended as very much self-service on the part of the customer, and is commonly set up as largely automated and repeatable. Therefore, a cloud service provider would need to adapt its sales and marketing processes accordingly.

Target markets need to be carefully evaluated and financial forecasts carried out to minimise risks and inform decisions. For these reasons, toolsets such as the SCF framework would be key in supporting these evaluation processes. Table 33 shows an example of the characteristic financial forecasts a service provider requires in evaluating a cloud investment, such as forecasting leads and customer volume estimations.

Table 33: Cloud SaaS Financial Forecasting

Annual Revenue Target	No. of New Customers	Annual Revenue per customer	Monthly revenue per customer	No. of users per customer	Cost Per Unit Per Month (PUPM)	Lead >Win Conversion Rate	New Leads Required
£1m	50	£20,000	£1,666	10	£160	20%	250

9.2.2 Operational Setup Considerations

The operational setup of a cloud service provider would need to meet the demands of this new business model. Some key elements that should be considered within this business setup are as follows:

- Roles, responsibilities and requirements for provisioning and deploying to new customers.
- Support and maintenance provision, and associated Service Level Agreements (SLAs).

- Billing setup, processes, frequency and payment terms. This could be fixed amount or pay-as-you-go for usage.
- Customer contract terms, length and cancellation policy. Examples being, monthly (cancel anytime), fixed term or evergreen (auto-renew).
- Contract terms and SLA's should reflect those underpinning the cloud service providers partner agreements. Example, if service provider has a dependency on a platform provider partner who offers a 1-hour resolution response, service plan should be to offer a 2-hour resolution in end-customer service agreements. This would allow an under-promise and over-deliver approach.
- Consider who does what in the provisioning process. This mean identifying what is the role and responsibility of the customer, providers depended upon, and the cloud service provider.
- Procedure for requesting support. Who would be taking the initial call and following through, the service provider or the platform providers.
- Raising a complaint and the issue resolution process.
- Account reviews with both customers and cloud providers depended upon, to ensure continued business.

9.2.3 Sales Structure Considerations

The service provider needs to consider the various requirements in the sales structure for their business, since the sales process changes within a cloud model. A sales team following a traditional approach would likely need to adapt to a new commercial reality. The following recommendations would support this transition:

- Sales commission models adapted for services over licenses.
- Complexity estimation for risk evaluation.
- Sales Training
    - Cloud is less about selling technology, and more about improving business. processes and employee productivity, reducing business costs.
    - New sales pitch for a possible different target customer base.
    - New sales materials and collateral.
    - Alternative sales cycle and customer decision journey.
    - Product training.

9.2.4 Key Performance Indicators (KPIs) and Benchmarks

In order to measure success of a new cloud portfolio once it is launched into the marketplace, there should be a series of performance indicators by which the service provider can measure success or identify areas for improvement. The suggested KPI and benchmarks listed below, should each have a SMART target (Specific, Measurable, Attainable, Relevant and Timely).

KPIs

- Cloud related revenue forecasts within a 6-month target.
- Periodic customer adds increases percentages.
- On-premises sales (X) vs Cloud/Hosted (Y) – Calculated as a ratio of X to Y.

Benchmarks

- Competition encountered:
    - Is a service provider winning deals that business would have lost without a cloud offering?
    - Are they competitive? How is performance against competition?
- Standard cloud solution deployments vs customised solutions. Calculate degree of customisation.

Other metrics

- New leads generated.
- Leads to Sales conversion rate.
- Pay-Per-Click (PPC) advertising costs vs performance.
- Business website and landing page traffic volumes.
- Marketing campaign activity.

9.3 A Cloud Transition Strategy

This research suggests that for complex business, the creation of a cloud strategy covering various solution offerings would provide for a more comprehensive, and less commercially risky, approach over a straight-to-cloud complex solution migration. A complex business would have specific niche requirements, so a strategic approach could mitigate some of the possible limitations evident in the cloud model. This would additionally allow the service provider to potentially gain a larger market penetration, within a shorter commercial time-frame. A good strategy would provide a wider choice to the service consumers, which would in-turn result in a larger market share. A target market checklist could be summarised as follows:

- Qualitative evaluation of the potential target market.
- Quantitative evaluation of the target market.
- Resource requirements and availability for deployment.
- Cost for channel penetration.
- Forecasting potential revenue and margins for SaaS.
- Demands and character of a cloud customer.

There are several options in pursuing a cloud strategy for both organisations and service providers. The SCF framework provides the means to more accurately quantify and validate the viability of these options detailed herein. The ideal strategy would be one that minimises the business risks, whilst maximising returns. Adapted for SCI from Wilson (2011), Figure 41 plots the expected increases in implementation risk as the complexity index the service provider aims to cater for, increases.

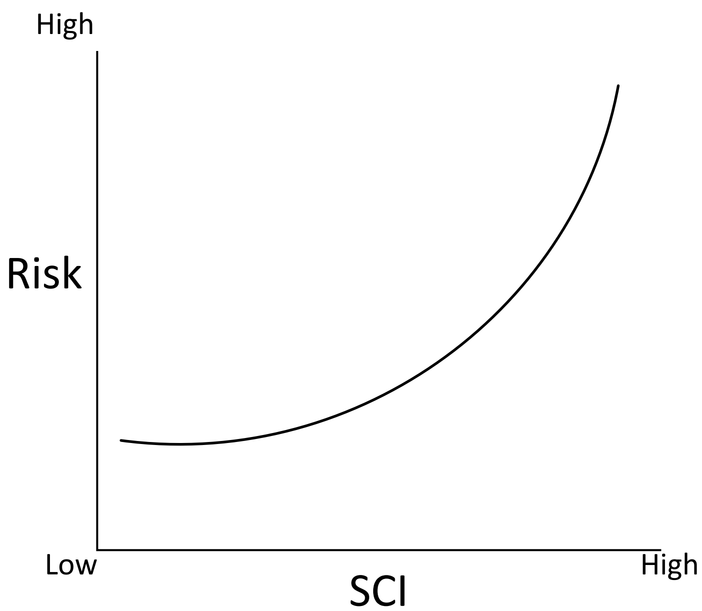


Figure 41: Graph-Risk vs SCI (adapted from Wilson, 2011)

For the service provider, transitioning the business to cloud services is not purely a technical exercise, but requires a more comprehensive commercial approach. To ensure a successful outcome for such an enterprise, this research demonstrates that planning needs to cover for the following key elements:

- Functional content.
- Pricing / Subscription / Administration / Licensing.
- Market identification and marketing.
- Sales process.
- Projects (Implementation & Deployment).
- Solution Cloud Architecture (R&D).
- Training (Internal).
- Documentation / Self-help content creation.
- Support setup.

9.3.1 Functional Content

Functional content is very much dependent on the other elements within cloud transition planning, such as the target market, and customer functional requirements. The SCF framework, as described in Section 6.3, could be applied within this context to map the functional requirements and processes for an industry, thus quantifying the risks and related commercial implications. The resultant data would in-turn inform the research and development requirements of the required business solution.

9.3.2 Pricing / Subscription / Administration / Licensing

The right pricing structures, and customer subscription forecasts, are a cornerstone of the strategy. These elements would define the success of the cloud transition as a commercial venture. Trends show that as complexity increases, the expenses incurred by the solution provider in managing and developing for this complexity also increases (Figure 45). This implies that revenues generated, and the underlying customer base, need to be of sufficient size to ensure a margin and return on investment. Adapted for SCI from Andrews (2014), the graph in Figure 42 suggests how price and customer subscriptions are expected to increase, initially at a higher rate and then levelling out as increases in complexity impact a smaller part of the market.

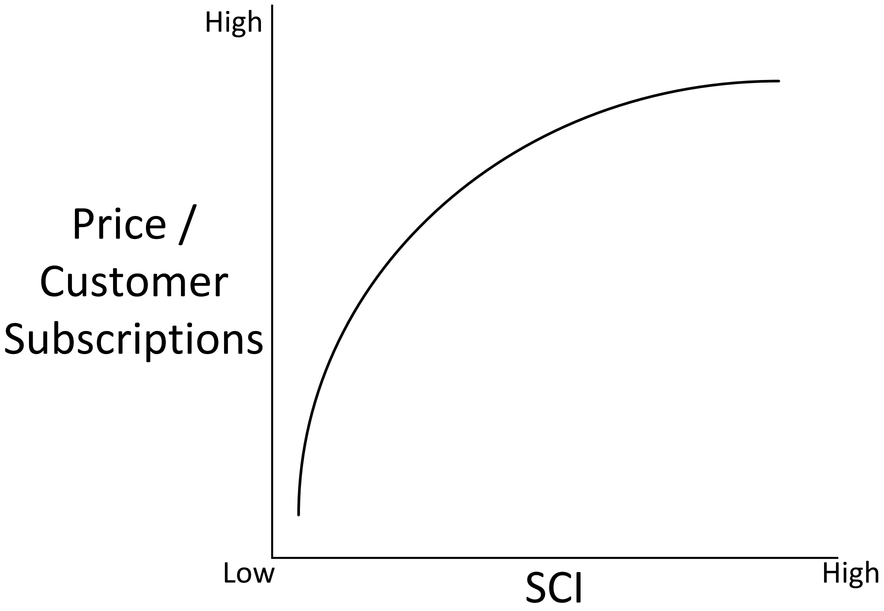


Figure 42: Graph-Price/Subscriptions vs SCI (adapted from Andrews, 2014)

Pricing for services determines charging structures, and revenue streams generated for the business. The following are some considerations for pricing:

- Pricing structures need to be defined to support channel models and subscription rates.
- Revenue structures should be in place to support modular pricing.
- Price structures defined for value-add services, such as month-end support.
- Price structures setup for one-time project cost within a subscription model.
- Administration strategies to retain control of hosting administration.
- Licences and modular access.

9.3.3 Market Identification and Marketing

The target market is the first business decision the service provider needs to make. Once the possible service consumers are identified, this allows for the analysis of their functional requirements, which would in-turn define the solution design. As detailed in Sections 1.6 and 1.7, there already is a considerable growing market for cloud based solutions. A holistic approach to market such services implies that cloud support could apply across all strategic options at various levels. This also ensures latent demand for a cloud based solution ahead of architectural R&D changes in developing a multi-tenant, single instance SaaS solution.

Within cloud services, the procurement decision is likely to involve the customer doing their own due diligence analysis, with minimal direct involvement by the service provider. Marketing therefore becomes a more important aspect of the sales process to ensure customers do not opt for competitor's solutions. An example of this is the provision of up-front pricing calculations for a subscription based solution, where the consumer calculates their own price based on elements like required modules and number of users, directly over the web.

9.3.4 Sales Process

The sales process for cloud services requires a different mind-set to traditional sales methods. As an example, the sales process relies less on the usual sales team structure and, in-turn commission based remuneration usually associated with these is less relevant.

For these reasons, a cloud transition plan needs to factor on-boarding of any existing sales team and partners, such as to present the cloud option as a key sales opportunity for them. This would need to be done alongside training in handling cloud related sales pipeline, and closing a quicker sale process.

Early adoption of this sales strategy could see cloud leads resulting in increased on-premises sales and vice-versa. This point becomes more relevant within the technical solution approach strategies proposed later in this chapter, since this sales strategy is common across all the proposed solution options.

Building up a re-useable sales approach, with key directed sales material, standard demo-approach, web chat questions and answers (Q&A), together with online sales videos would go a long way to ensure the potential customers subscribe to the service.

9.3.5 Projects (Implementation & Deployment)

A robust project methodology, to incorporate cloud services within traditional implementation methods, would serve any existing customer base and yet allow reusable concepts within the transition from on-premises to hosted.

Some such re-useable elements could be:

- Re-useable implementation materials, example: rapid start templates, self-help documentation, set deployment time-scales, data migration templates, online training videos.
- Consultant focus on configuration rather than customisation.
- Defined, and shorter implementation processes.
- Package key tasks such as data migration.
- Develop structures for extensive remote deployment and implementation options.

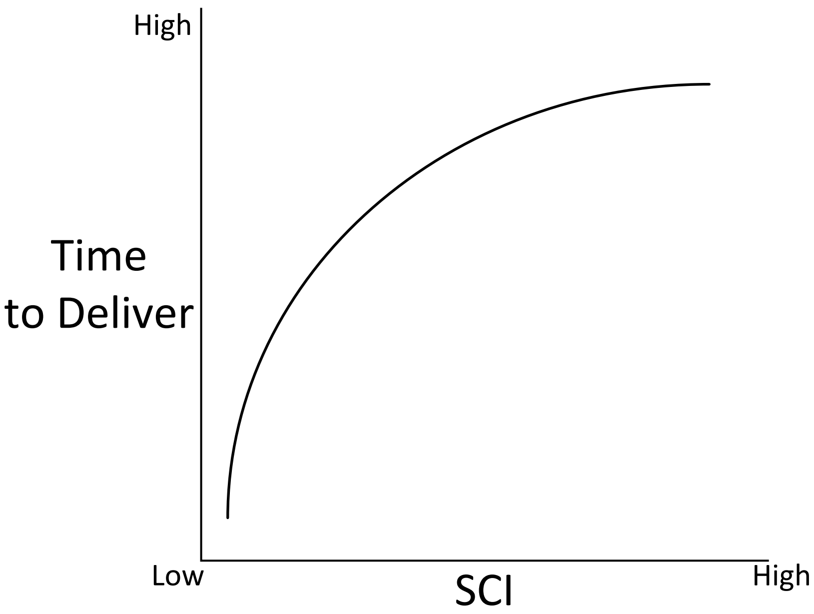


Figure 43: Graph-Time to Deliver vs SCI

Informed through the author’s industry experience, the graph in Figure 43 represents the view that the more complex the system is, the longer it would take to deploy, both from a SaaS service provider perspective and from a business implementation standpoint. Having the methodologies described above in place, would ensure that time-to-deliver should equalise eventually resulting in higher complexity having only marginal higher rates of implementation time requirements.

9.3.6 Training (Internal)

Training considerations are important to ensure that internal resources within the service provider business are prepared. The cloud strategy would result in direct and indirect changes, which would provide new challenges to the business. Stakeholders within such training exercise would be the sales team, business partners, support team, delivery team, development team, and the like.

Such a training exercise, within a new or changing business would become inevitable as the technology progresses. Thus, it would pay-off to be proactive in this, increasing the organisation's responsiveness within the market.

9.3.7 Documentation / Self-Help Content Creation

Cloud services transition away from traditional solution deployment models, with the customers being more proactive in their understanding of the solution, and their consumption of related services. In view of this, ensuring provision of reusable help documentation, white-papers, self-help videos and online tutorials, is key in allowing current and potential customers to manage their own product familiarisation and decision-making process. This holds true, both before and after the sales process.

Such training functions also serve mitigate the load on the support team and help-desk in dealing with more generic queries. The service provider could then better focus on value-add services, such as additional consultancy services, and specific surround requirements, to mention a few. This approach additionally provides for the possible generation of side-revenues, through a structured customer certification process, built within the cloud strategy.

9.3.8 Support Structure Setup

A solid support structure, with re-useable methodology for pre-sales and after-sales helpdesk customer interactions in critical in cloud services. This ensures customers keep using the service provider's own services, and not move to competitors. Additionally, within a single instance, multi-tenant system, an issue identified by one tenant could in effect impact the whole customer base. Identifying and resolving system issues quickly in this case is paramount.

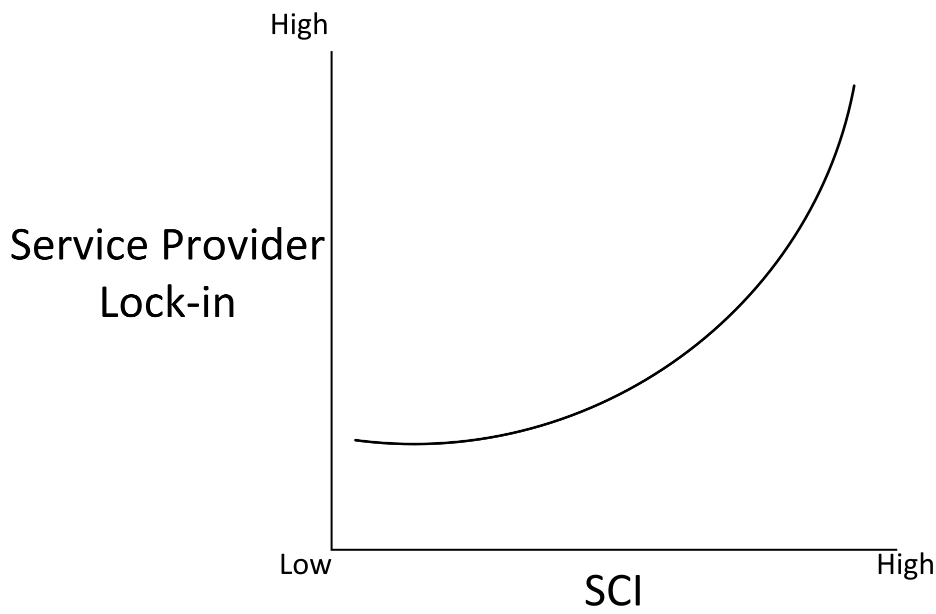


Figure 44: Graph-Provider Lock-in vs SCI (adapted from ITSM Portal, 2010)

The support and helpdesk team could have an extended role in support of potential customer purchase decision process, Q&A prior to purchase, possibly through online chat. Additionally, purchase could be enforced through subsequent post purchase support, facilitating value-add service sales.

Informed by ITSM Portal (2010), vendor lock-in review within Cloud, Figure 44 represents the risk, that a higher complexity in the business and supporting solution, makes it more difficult to switch service providers. This however could be considered a benefit to the service provider, as it ensures a lower churn rate, and higher subscription retention rates.

9.3.9 Research & Development (R&D) - Product & Platform Architecture

Arguably, the most significant expense within a cloud strategy is the solution architecture design and development. The author suggests that separating this R&D element from the other surrounding cloud transition elements, enables a plug and play approach for the various levels of cloud integration and platform strategy options.

This strategy of separating R&D would be particularly useful to mitigate cloud platform risks and delays, pending platform providers' own technology releases. In this manner, it allows a proactive, rather than a reactive, approach to the transition. Thus, updates to cloud technology do not delay the overall cloud strategy, allowing possible cloud based sales revenues without the full R&D expense. The possibility of separating decisions around R&D spend, whilst still gearing up the business to embrace cloud solutions, is another key advantage, and would likely result in a higher business valuation for the service provider with its investors. Adapted for SCI from the Product Life Cycle stages by Claessens (2017), Figure 45 outlines a representation of how the investment cost in a SaaS system compares to the business complexity it services. This is an effective barrier to entry for potential competitors, and is even more evident than with traditional software selling models. On the onset however, there are still costs that would need to be incurred, relatively independently of complexity, such as employee training, infrastructure and other elements represented within this chapter.

Evident within generic SaaS models is that whilst there is the possibility of a wider market, and the increased revenues that would bring, the initial take-up of the solution by consumers is relatively slow. A consideration is that the service provider is incurring increased start-up costs, with limited initial revenues to mitigate. This entails that it would take some time to see a return on investment, or even to breakeven. The upside however is that with a complex SaaS aimed at a wider customer base, the potential returns would far outstrip those of traditional software selling models. Following this initial cost, the ratio of investment against complexity is then expected to show a steady rate of increase, resulting in a higher cost for higher complexity. This rate of increase is then expected to decrease as the solution matures and the service provider effort transitions primarily to maintenance.

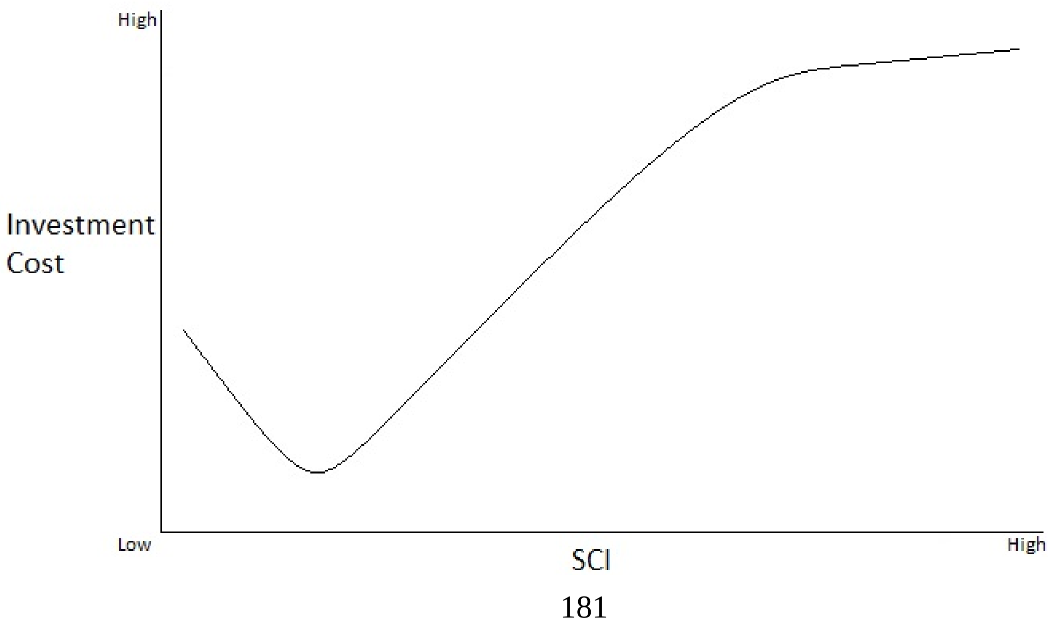


Figure 45: Graph-Cost vs SCI (adapted from Claessens, 2017)

As this research examined the trends in the ERP industry, it finds that migration to cloud architectures for ERP will inevitably occur, with limited options for alternatives to the customer. Indeed, early adopters are observed from recent professional experience. Carr (2017), analysed these trends, and their impact on legacy systems. The determination is that quite likely, with newer releases of proprietary solutions, the provision of some degree of cloud deployment options for traditional service providers would thus be inevitable. It therefore becomes a case of WHEN rather than IF, for the service provider not to lag the competition. Product development methodologies could therefore be defined on the onset, to reflect common requirements within each platform architecture. This should in-turn lead to a more agile approach to product releases and hotfixes, in line with expectations for multi-tenant environments.

9.4 Definition and Analysis of Cloud ERP Service Options

This scenario provides a strategy allowing a service provider to position their business across cloud hosted services, rather than solely with on-premises systems. This research has identified several strategies that could be applied when considering such a strategy. This example is intended to provide options for extra revenue streams to complement a service provider's traditional core business.

When referring to Cloud, there is an appreciation for the benefits of having redundancy, backup capabilities, security and scalability through services such as Azure & Amazon EC2.

The Cloud would emphasise the alternative approaches that are needed for traditional 'Projects' as compared to 'Products'. The key focus for a cloud solution is to enable organisations to adapt the business rather than the solution. This would require a change in mindset for some service providers, to help customers adapt their processes rather than the service provider customising the 'Product'.

The following are hosted service options this research identified as currently available for a service provider:

Table 34: Cloud SaaS Transition Strategy Options

No.	Strategy	Description	Key Characteristics	Estimate SCI
1	Single tenant legacy solution hosted on Cloud.	Deliver legacy ERP solution hosted on the Cloud	Multi-Instance / Single Tenant	120 SCI (Max)
2	Multi-tenant legacy solution hosted on Cloud	Deliver legacy ERP solution hosted on the Cloud with a multi-tenant SQL backend	Multi-Instance ERP / Single instance SQL / Multi-Tenant	96 - 120 SCI (Avg – Max)
3	Complex SaaS solution	Provide a SaaS complex business solution as SaaS	Single (or Multi) instance / Multi-tenant	96 SCI (Avg)
4	Lite SaaS solution	Provide a simplified SaaS business solution as SaaS	Single (or Multi) instance / Multi-tenant	71 SCI (Min)
5	Targeted companion apps	Targeted SaaS apps that complement an on-premises solution	Single instance / Multi-tenant	1 – 13 SCI (1 SCF Process Node)

9.4.1 Option 1: Single-Tenant Legacy Solution Hosted on Cloud.

Legacy ERP solution hosted on Azure/EC2 Cloud, Multi-instance / Single Tenant.

Table 35: Strategic Option 1 Summary

SCI (Estimated)	120 SCI (Max)
Time to Market	Low
Cost to Market	Low Cost
Risk	Low

This option would take existing product offerings and offer these as a hosted cloud solution with option of subscription pricing. As the ERP solution is still tailored for a single tenant, then this strategy could support the maximum business complexity of 120 SCI (Table 26).

The legacy solution would be provided on a set of supplied virtual servers on Azure, EC2 or equivalent. This would remove the need for a customer to have their own servers and reduce the hardware and support overheads.

As an option, this can be offered very quickly with little impact on core service provider services and their existing customer base. Additionally, it allows for easy expansion within the existing market, providing additional service options to customers that might be looking for alternate ways to fund their ERP deployment through subscription pricing.

A legacy ERP hosted on the cloud, offers customers a platform, which is scalable, secure, and performance-efficient with the option of flexible subscription licensing, without compromising on functionality or flexibility of customization to meet business needs.

- Microsoft Azure / Amazon EC2 provides scalable, secure, performance-efficient storage services in the cloud.
- Flexibility: subscription allows for easier migration into the cloud space. It turns a once rigid on-premises ERP system into a more flexible tool, enabling the customer to use as much or as little space as required, scaling their usage as their business grows.
- More configuration options: Rather than offering a “one-size-fits-all” approach that can translate to “one-size-fits-none”, it offers flexible options for each customer, including customisations.

Advantages:

- Quick and easy.
- Ability to re-sell platform services, which would act as a continued ‘new’ revenue stream.
- Customer is not required to make capital cost expenditure on hardware.
- Easy to customise.
- Per-customer customisations still possible.
- Scalable.

- Allows a choice for customers to purchase as a subscription based license or traditional one-time license model.
- No compromise on functionality or customisation options.

Disadvantages:

- Increased cost of ERP solution for customers that already have hardware.
- Not a LITE option, provides full ERP functionality, even where not needed.
- Pricing models for platform services are currently quite complex.

Table 36: Strategic Option 1 Implications

Deployment Options	Enterprise / Professional / Rapid
Target	Enterprise / Large
Implementation Type	Project
Sales Process	Traditional Onsite Sales
Licencing	Up Front or Subscription
Training	Onsite / Remote
Business Process Consulting	Traditional
Data Migration	Traditional
Customisations	Possible
Industry Solution	Yes
Surround Technology	Yes
Go-Live Support	Traditional
Support	Support Contract

9.4.2 Option 2: Multi-Tenant Legacy Solution Hosted on Cloud

Deliver legacy ERP solution hosted on the Cloud with a multi-tenant SQL backend.

Table 37: Strategic Option 2 Summary

SCI (Estimated)	96-120 SCI (Avg-Max)
Time to Market	Low
Cost to Market	Low Cost
Risk	Low

This option would take existing product offerings and offer these as hosted cloud solutions. Main difference being the use of multi-tenancy options of more recent SQL server versions for their database back-end. The legacy solution could still be provided on a set of supplied virtual servers, like Azure or EC2, so providing the same advantages as that of option 1, however the facility for a hybrid multi-tenancy solution would further reduce costs, and make for a more competitive offering. The ERP solution could still be tailored for a single tenant; however, the multi-tenancy element of the back-end could impose some limiting considerations on the general complexity, which would have parallels with those found in a direct SaaS solution. In view of this, the estimated business complexity this strategy option could support would range between the average 96 SCI and the max 120 SCI, (Table 26).

This option can be offered relatively quickly, within the same time-frames as those of traditional ERP deployments. This would have marginal impact on core service provider services and their existing customer base, though the ramping-up for a multi-tenancy database setup would require additional investment in training.

Figure 46 lists the patterns appropriate for each of the three approaches, divided into sections representing the three SaaS qualities of security, extensibility and scalability.

- The security patterns listed below demonstrate the design for an application with “virtual isolation” through mechanisms such as permissions, SQL views, and encryption.

- Configurability allows SaaS tenants to alter the way the application appears and behaves without requiring a separate application instance for each individual tenant.
- The extensibility patterns describe possible ways the data model can be implemented such that tenants can extend and configure individually to meet their requirements.

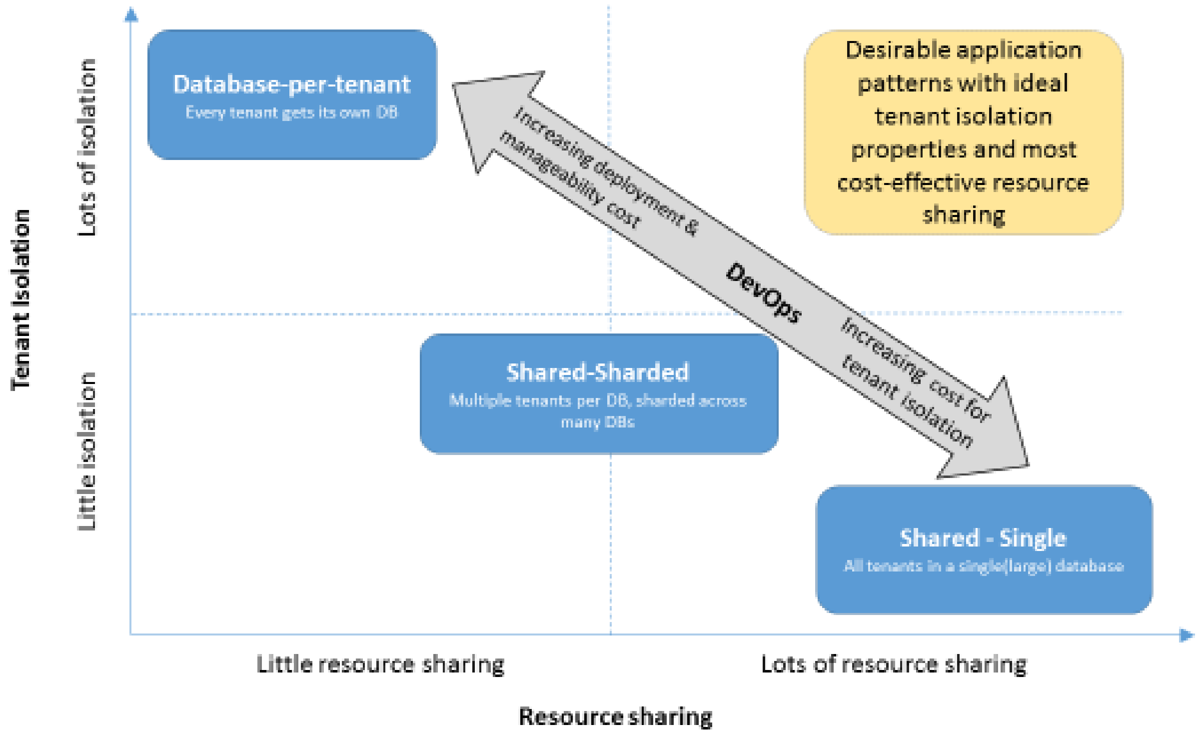


Figure 46: Multi-Tenancy Development Models (Microsoft, 2016)

Advantages:

- Rapid and de-risked deployment with a standard product.
- Pay-as-you-go (subscription).
- Ideal for smaller operations.

Disadvantages:

- Build effort is required support multi-tenancy database back-end.
- Training and support materials all required online.
- Pricing models need factor-in payment for database, as well as hosting services.
- Requires a stable product.
- More limited customisation options then option 1.

Table 38: Strategic Option 2 Implications

Deployment Options	Enterprise / Professional / Rapid
Target	Enterprise / Large / Medium
Implementation Type	Project
Sales Process	Traditional Onsite Sales
Licencing	Subscription
Training	Project
Business Process Consulting	Project
Data Migration	Project
Customisations	Possible / Limited
Industry Solution	Yes
Surround Technology	Yes
Go-Live Support	Traditional
Support	Support contract

9.4.3 Option 3: Complex SaaS Solution

Develop a full SaaS solution covering the business complexities of the target market.

Table 39: Strategic Option 3 Summary

SCI (Estimated)	96 SCI (Avg)
Time to Market	Long
Cost to Market	High Cost
Risk	Higher risk

This is the most complex SaaS solution, having the SaaS product covering the requirements of the target market more comprehensively, with key process complexities handled directly by the solution as standard. This solution would require a good understanding of the requirements of the target market, and would require a considerable development budget to build the solution.

As described in Section 5.2, the more complex the target market and related businesses, the more complex the solutions required to support them. This implies that the solution becomes more targeted, which would be a competitive advantage for businesses within that market, yet limits the size of the market that could make best use of that solution. This potentially translates to additional business risk for the service provider and highlights the importance of choosing the right market, for the right investment returns. In this respect, the SCF framework could be invaluable in quantifying these elements scientifically. The application of this option to the business case results in Chapter 8 would suggest that the industry average of 96 SCI (Table 26) would roughly provide the ideal market coverage for the investment made in this example.

Advantages:

- Competitive advantage within target market and barriers to entry for competitors.
- Captures larger portion of revenues, since customers would make more use of the one solution, rather than involve other third parties.
- Less customisation requirements, so more likely customers taking up the SaaS option.
- Complex SaaS solution would provide additional parameterisation control, and therefore additional flexibility.

Disadvantages:

- Higher research and development costs.
- Higher business risk in recovering investment.
- Longer term returns forecasts.
- Bigger upfront cost outlay.

Table 40: Strategic Option 3 Implications

Deployment Options	Enterprise / Professional / Rapid
Target	Enterprise / Large / Medium
Implementation Type	Product
Sales Process	Remote
Licencing	Subscription
Training	Remote
Business Process Consulting	Hour/Day (T&M)
Data Migration	Hour/Day (T&M)
Customisations	None
Industry Solution	Yes
Surround Technology	None / Limited
Go-Live Support	Hour/Day (T&M)
Support	X No. of Incidents

9.4.4 Option 4: Lite SaaS Solution

Develop a cut down SaaS solution, simplifying the business complexity of a target market, whilst covering main business processes.

Table 41: Strategic Option 4 Summary

SCI (Estimated)	71 SCI (Min)
Time to Market	Long
Cost to Market	High Cost
Risk	Medium

This option would result in the development of a cut-down SaaS solution that simplifies the processes surrounding a complex industry, possibly through supporting linear best practice scenarios. This would require less R&D effort to the provision of a full solution. Investment requirements would also be less, though still not negligible. This solution would be directed at those businesses that are able re-engineer their business process to reflect the simplifications within the solution.

In application to the business cases, the results would suggest that an SCI of 71 would be the ideal complexity coverage point for this option. This value being the minimum complexity support recommended for that industry, as per Table 26. The key characteristics of this strategic option are as follows:

- Lite standard solution using latest SaaS technology incorporating simplified best practice processes.
- Latest releases. All customers would share the latest release of the product incorporating latest functions.
- Flexible subscription provides a scalable solution that is fast to respond to customers' needs.
- Solution would support a single instance, full multi-tenant cloud solution. Separate instances could still be supported for those customers that require additional security and control over the system.
- Strategically, this could serve as a stepping stone for the service provider, growing the solution over a time-period to a full solution.
- The business setup required by the service provider in support of such a solution, would not differ from the Full SaaS solution option. This applies to marketing, pricing, training and support, amongst others, as discussed in other sections.

Advantages:

- Rapid deployment supported.
- Fast to deploy new functionality.
- Access to industry market, mitigating some of the risk.
- Next generation of software deployment.
- Early to adopt, creates foothold in specific markets.
- Strategically supports future expansion of solution to a full SaaS option.

- Allows service provider to focus on setup of surrounding business support structures, such as marketing and training.
- Proper SaaS - single instance / multi-tenant approach.

Disadvantages:

- Considered a longer-term proposition.
- Still a complex option requiring R&D investment.
- Requires a pre-configured solution prior release.

Table 42: Strategic Option 4 Implications

Deployment Options	Enterprise / Professional / Rapid
Target	SME & Micro
Implementation Type	Product
Sales Process	Remote
Licencing	Subscription
Training	Remote
Business Process Consulting	Hour/Day (T&M)
Data Migration	Hour/Day (T&M)
Customisations	None
Industry Solution	Yes
Surround Technology	None / Limited
Go-Live Support	Hour/Day (T&M)
Support	X No. of Incidents

9.4.5 Option 5: Targeted Companion Applications

Target SaaS apps that complement an On-premises solution. Single instance / Multi-tenant.

Table 43: Strategic Option 5 Summary

SCI (Estimated)	1-10 SCI (Single Process Node)
Time to Market	Medium / Low
Cost to Market	Medium / High Cost
Risk	Medium

This option approaches the development of a SaaS solution through the provision of companion applications (apps), providing targeted solutions to specific industry functions. This would require a more granular approach to the SaaS solution being developed by the service provider, with the possibility of stringing together various apps to within an online marketplace. An example of such an approach could be Microsoft Office 365, or Google apps (Figure 47), which provide access to various apps and add-ons within a web marketplace. Businesses could then take on as many apps as required to support their own processes.

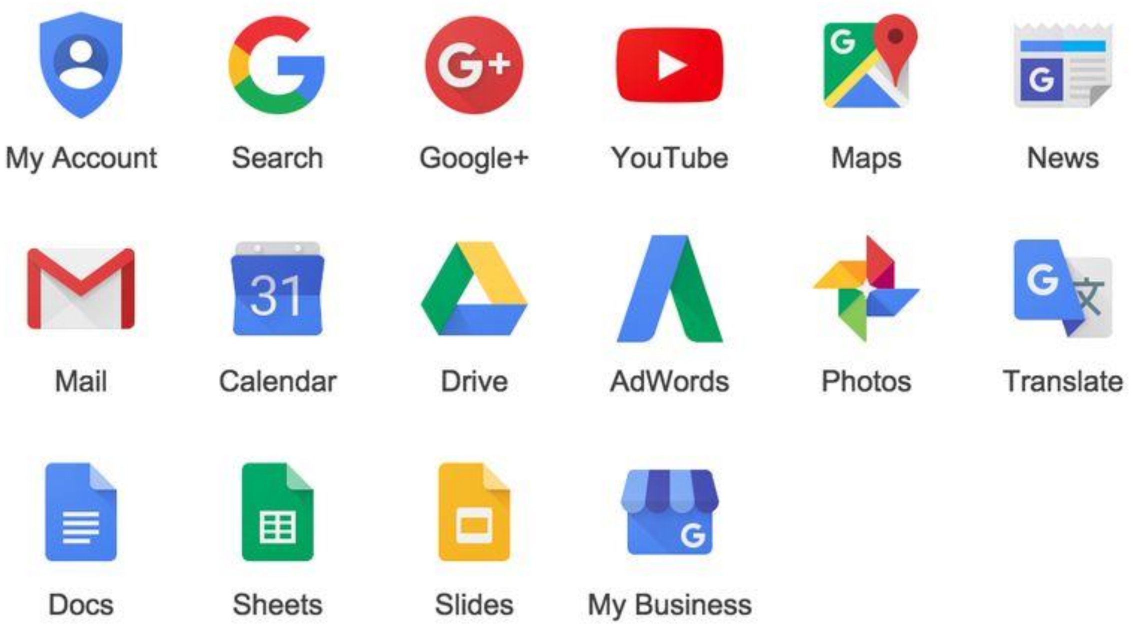


Figure 47: Google Web Apps (Google, 2017)

An app could provide functionality limited to simply to the requirements of one SCF Process Node. The business complexity coverage would therefore be relatively low. Business case results suggest that an SCF Process Node could have an SCI range of 1 to 13 in this scenario, as summarised in Table 26. The key characteristics of this approach being:

- Minimises the time to market for a service provider solution.
- Granular functionality implies the requirements of a wider market base could be met.
- Apps could complement existing on premises complex business systems, enabling take-up by any existing customer base.
- Allows possibility of interfacing together several apps to cover more complex processes.

This approach additionally enables new service providers to gain hosting experience, and lessons learnt from the original exploration into cloud services and apps would act as good grounding into the introduction and deployment of more complex solutions. It could complement existing on-premises business solutions the solution provider might already provide, serving as a mechanism for bringing new customers to its legacy solutions.

With this strategy, it would be imperative to be able to incorporate new SCF process nodes within the app as it evolves, or alternatively stacking together additional apps. This allows existing customers to continue using the software as is, or to embrace the newer releases with their additional functionality and bug fixes.

Advantages:

- Minimizes risk and investment.
- Enables business to gain experience with cloud services.
- Could complement legacy on-premises solutions.
- Could be expanded to other apps.
- Potential for a wider market.
- Same solution for all, single code base/multi-tenant.
- Potential for freemium approach to pricing (In-app purchases).

Disadvantages:

- Still requires build-up of business skill sets.
- More generic solution implies possibility of more competition from other apps.
- Different app stores, support alternative technologies.
- Expanding to fully support a complex business would be more difficult, so as not to impact current customer base.

Table 44: Strategic Option 5 Implications

Deployment Options	Rapid
Target	SME & Micro
Implementation Type	Product / App
Sales Process	Remote / Online
Licencing	Subscription / Freemium
Training	Remote / Online
Business Process Consulting	None
Data Migration	None
Customisations	None
Industry Solution	Yes
Surround Technology	N/A
Go-Live Support	X No. of Incidents
Support	X No. of Incidents

9.5 Analysis and Summary of the Service Delivery Options

Analysis of the five viable options described within this chapter, identified complex business implications for each option. Based on an evaluation of these, the author concludes that a less risk averse strategy would be to potentially take a three-phased approach, which includes the implementation of all the strategies across a defined time-frame. Supporting this outcome, Figure 48 outlines a summary of the various SaaS strategy options, whilst Figure 49 represents the various strategy options compared through a magic quadrant.

No.	SaaS Strategy	SCI	Time to Market	Cost to Market	Risk
1	Single tenant legacy solution hosted on Cloud	120	Low	Low Cost	Low
2	Multi-tenant legacy solution hosted on Cloud	96 - 120	Low	Low Cost	Low
3	Complex SaaS solution	96	Long	High Cost	High
4	Lite SaaS solution	71	Long / Medium	High Cost	High
5	Targeted companion apps	1 - 13	Medium	Medium Cost	Medium

Figure 48: SaaS Strategy Summary

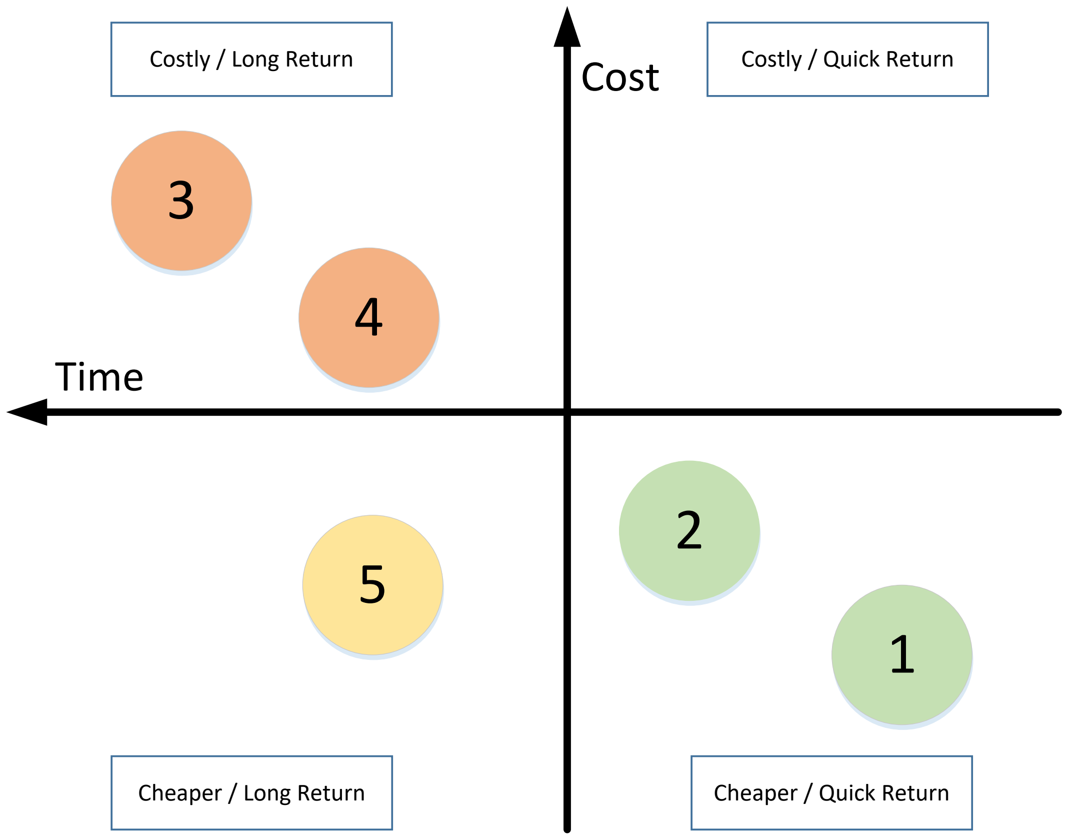


Figure 49: Strategy Options Quadrant – Cost vs ROI

In this case, an initial phase 1 could be to provide a cloud offering which includes Option 1 & Option 2, as these can commence in unison. Both options can be delivered relatively quickly and at a lower cost.

To complement option 1 & 2 within phase 1, work on option 3 can be initiated in parallel, since this would require an alternative skill-set, and therefore likely a separate resource set.

Phase 2 for the provision of a cloud solution, would then include taking the lessons learnt from phase 1 and deploying Option 3. The timescales for option 3 are lengthier, so to ensure business momentum and interest in the service provider's cloud products, subsequent enhancements to option 3 can run in parallel. This may take the form of industry specific companion apps, or more generic offerings across multiple platforms.

Phase 3 for delivering a comprehensive cloud offering would see the extension of the existing option 3, to cover for the full complex business requirements of the target market. This cloud transition strategy is represented in a graphical timeline, in Figure 50.

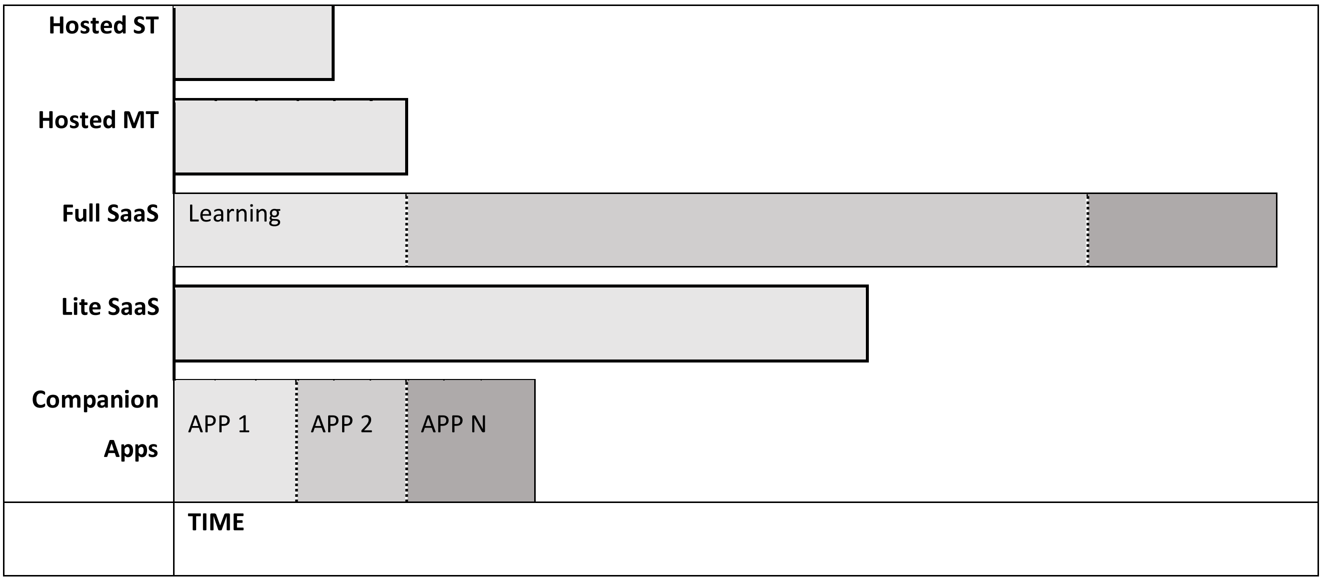


Figure 50: Cloud Transition Strategy Timeline

9.6 Common Elements within the Cloud Strategy

The overarching ERP cloud transition strategy, proposed by the author in Section 9.5, recommends that a service provider incorporates elements from all the five options into one transition plan for a defined timescale. This strategy would yield several benefits, such as minimising risk and spreading costs over a structured growth period. This is evidenced by taking advantage of the common elements across the five options. Figure 51 shows a common planning approach for these various cloud options, which defines the elements as modular tasks within an implementation plan. As a result, this would gain time and cost-savings within the business, whilst ensuring robust methodologies are in place, in preparation for the transition to a complex SaaS solution.

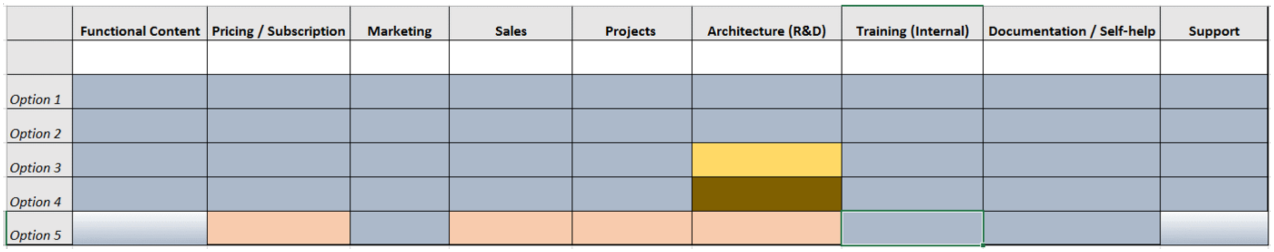


Figure 51: Cloud Strategy Commonality

The colour key for Figure 51 is as follows:

- **Same colour** - indicates that applying a structured, holistic approach to individual elements implies the same output could apply across two or more elements.
- **Partial colour** – partial colour match indicates partial commonalities across elements.
- **Different colour** – this indicates no commonality across elements, and are therefore specific to that strategic option.

Such an overview serves to highlight how building these common elements as separate components, each having their own implementation strategy, allows better control of the costly R&D investment within the wider cloud strategy. Other than solution development, in this case R&D would have a lower impact on the overall plan, and cloud transition timeline.

The resultant cloud strategy being presented here is directed at service providers aiming for a full transition to the cloud as their main service model. Elements of the five strategic options can however still be applied, in part or in whole, by service providers interested solely in having a cloud presence, whilst retaining their commercial interests based on traditional deployment models.

9.7 Extended Multi-Tier Models as a Solution to Complex SaaS ERP Systems

The investigation in Chapter 4 found that such strategies are closely linked to the underlying computing architectures of the ERP system. The author therefore deduced that for a complex business system to be better adapted to the SaaS model, business complexity needs be broken down in smaller and more manageable components, which can then be linked together again to form the whole solution. The system architecture should therefore allow various parts of the solution to be provided as web services and hosted by one or more integrated service providers. In practice, this would even allow for parts of the solution to be retained on-premises. Such web services could track the SCF process node and process functional elements of the solution, enabling functional low-level deployment flexibility and scalability. This proposed technical architecture approach would thus mitigate some of the limitations identified at the onset of the research, with traditional ERP system architectures of the time.

Over the duration of the research, the validity of this deduction was evidenced by the direction the main industry players have taken up, as the multi-tiered, distributed model continues to evolve in complexity, though conceptually remaining the same. Release iterations of traditional ERP systems have shown a continuous restructuring of the underlying solution architecture, enabling increasing levels of cloud integration and support. This evolution is still ongoing at the time of writing of this thesis, and progressing steadily as ERP technology draws nearer to the cloud rhetoric.

One such example could be the AX ERP system. The diagram represented in Figure 52 shows a high-level overview of the system architecture for this ERP system, within one of the more recent releases. Note how the architecture is broken down into a myriad of components, allowing flexibility in the underlying infrastructure. This infrastructure could consist of many of these ERP components, with these components able to be installed on either single or multiple physical servers. For some components, these could be hosted on both physical or virtual servers. This approach paves the way for more flexibility in the provision of the solution within a cloud hosted model.

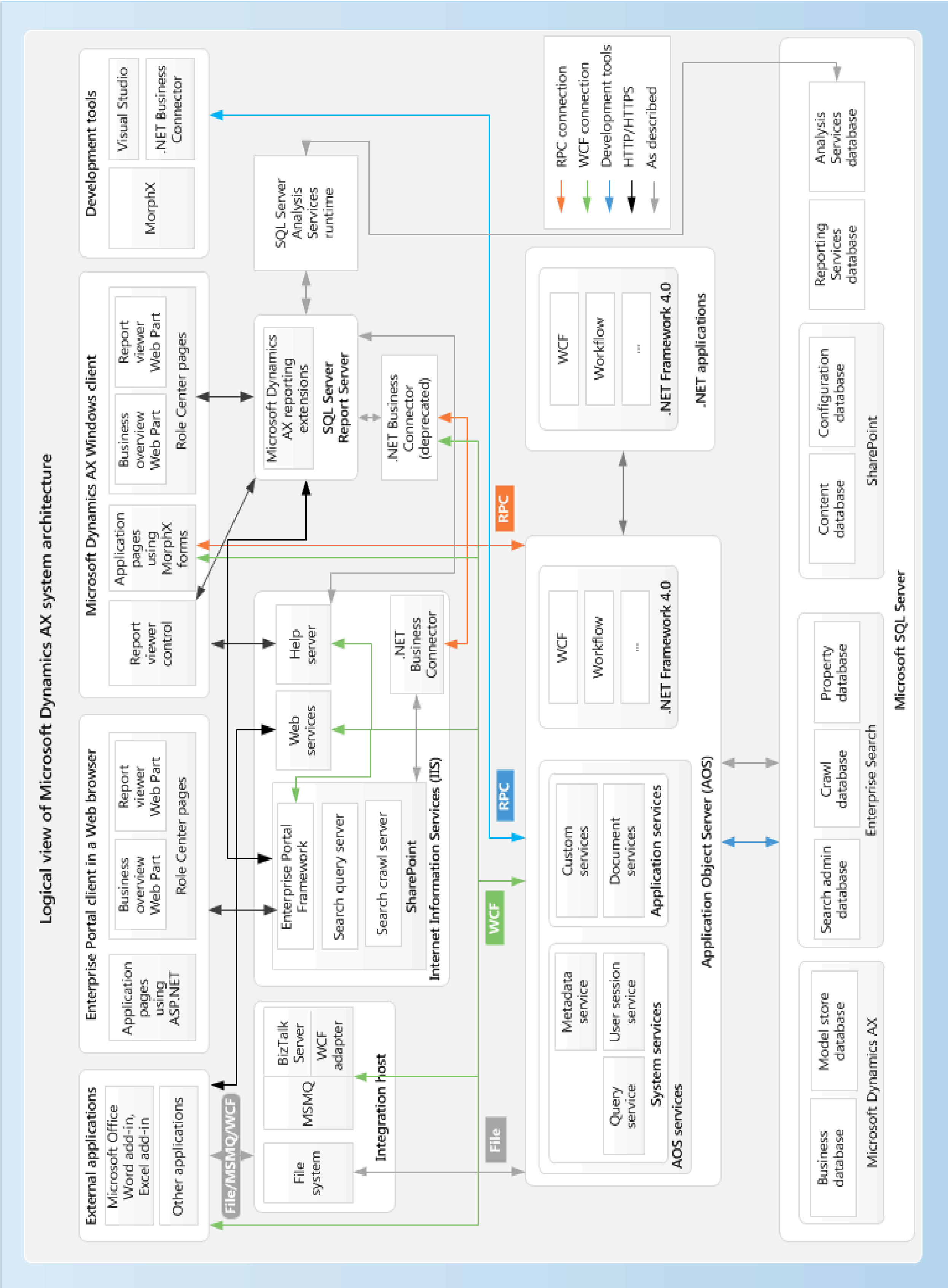


Figure 52: Example of Extended Multi-Tier Architecture (Microsoft, 2016)

9.8 Summary

This chapter expanded on the results derived from the business case application of the research derived complexity framework, to collate a cloud transition strategy for the cloud service provider. In Section 9.4, various strategy options were proposed and analysed, both individually, and then holistically, applying the SCF framework results from the business cases for their SCI complexity evaluation. The risk and ROI elements for each strategy were evaluated, and considered within the wider commercial elements of the possible target markets described in Section 1.7.

Section 9.6 then described how an overarching cloud transition strategy that considers the commonalities between the various strategic options would result in a faster transition. The author concluded that a holistic cloud transition strategy that covers the five range of options is therefore recommended, and would reduce the business transition risks and costs, whilst delivering a cloud presence within a shorter time-frame.

Consideration was additionally made of how these strategies are directly influenced by the underling computing architectures. Section 9.7 discussed the deductions made on the required evolution of these computing architectures, to better support complex cloud ERP. Based on this analysis, the author extrapolated upon these architectural trends to propose a view on a future architectural model, structured as ERP web services. This provided a forecast on the general direction the ERP industry could be evolving to. It was also noted across the duration of this research that elements of this forecasted architecture were validated in newer ERP releases by key industry players.

The strategies proposed here-in have been primarily informed through the author’s practical experience, and as such will need to be validated through their practical application in future work.

CHAPTER 10

CONCLUSIONS AND FUTURE WORK

10.1 Introduction

The main motivation behind this thesis has been to research complexity within business and its impact on the transitioning of business entities, particularly those with complex processes, to cloud hosted solutions. The results from this work have validated the research hypothesis outlined in section 1.5, showing how a business could be modelled and accomplishing the quantification of complexity in support of strategic analyses within decision making.

This chapter summarises the contributions of this thesis to this field of knowledge, and explores the possible future directions from this line of investigation.

10.2 Research Summary

As cloud solutions are being represented through various sales and marketing rhetoric, comparison has been found as generally limited to the subjective views of industry analysts. Traditional solution deployments of complex business systems, such as ERP software, rely heavily on the service provider bridging this gap between business processes and software solutions. This would generally involve an in-depth analysis of that business, with the service providers' consultants applying their intimate knowledge of proprietary ERP solutions to the requirements of that business, within a capital expenditure project.

The cloud business model however is forcing a change to this traditional approach. Though this cloud business model is somewhat still in flux; from observations in practice, the conclusion is this outcome is inevitable. This leads to an increased focus on the service consumer being more proactive in analysing prospective solutions, and minimising the involvement of the consultant middleman. The service provider, in turn, would be increasing focus on marketing a common solution to a wider "generalist" audience, with service consumers buying into the solution as a service, and as an ongoing operational expense.

This implies that due to this switch in focus, a business transitioning to the Cloud would need be more proactive in garnering an in-depth understanding of the various SaaS solutions on offer, effectively bridging those solutions to their business processes internally. This would be a key step in the strategic decision-making process, to ensure that the most compliant solution is selected at that point of business operations, for the right cost.

This thesis provides a new framework to facilitate complex business modelling to cloud business solutions, offering tools that contribute to the efficiency and effectiveness of their deployment. This framework could similarly be applied to reverse engineer requirements by ERP service providers modelling a business or industry.

The significant contributions made by this research are as follows:

10.2.1 A New Definition for Business Complexity

This research determined that a better understanding of business complexity is key to ensure a better translation of business requirements in support of ERP as a SaaS solution, detailed in Section 5.3. A distinction is made between complex business systems and simpler ones, as this was discovered to have a direct impact on the ease of transition to a cloud hosted model, as outlined in Section 3.2. The literature reviews identified dichotomies in the meaning of complexity, which varied widely depending on the context this was applied to. Additionally, existing business complexity definitions in Section 5.2, were found not to be directly applicable to the requirements of the SCF framework, as these were found to cover the wider business landscape, whilst excluding the underlying supporting systems and technology elements. The author noted in practice that a business would only have direct control on its own structures and underlying systems. Accordingly, these were the elements the framework was directly focused on.

Business Complexity Definition: This work derived its own definition of a complex business, this being one that covered as many of the various business processes described in Chapter 5, with these operating inter-dependently within its system functions.

Referencing ERP systems as exemplars of complex business systems. This research observed that an ERP implementation will not necessarily imply a complex business system, since partial modules can be implemented separately to cover only particular aspects of the business. Therefore, applying the author's definition of business complexity, a complex ERP system deployment would be one that demonstrates that most of these functional elements are in operation within that business environment.

As a result, the author determined that business complexity would increase further if disparate systems and technologies were used to handle the various business elements, as this would traditionally require data interfaces, mapping structures and communication protocols, with some of these potentially being provided as services through third parties having their own proprietary systems.

10.2.2 A New Framework for Modelling Complex Business to Cloud ERP

This research has introduced a new framework for business modelling, the SaaS Complexity Framework (SCF) in Chapter 5. SCF seeks to provide a scientific means for business complexity to be measured and modelled across cloud ERP solutions. A principal innovation driver for this framework was its intended use to inform modes of migration of ERP within a cloud transition strategy. This framework has additionally been found to be similarly effective within more traditional solutions deployment models. The main features of the framework being as follows:

10.2.2.1 *A New Metric for Calculating Complexity in Business*

The new SaaS Complexity Metric (SCM) that has emerged from this research was developed with the aim to synthesise a complexity estimate measure for a business. Section 5.5 details how this was derived from the number of specific processes and functional elements that an organisation employs. The SCI infers a singular relation between how complex the organisation is and the number of interrelated process elements that contribute its business. This definition is extended to suggest that the resulting complexity of underlying supporting systems is therefore directly correlated to the complexity of the business itself, for example as shown in Figure 21 and Table 23.

10.2.2.2 *A Unified Model Enabling the Business Mapping Process*

A key element of the SCF framework was the development of a system-agnostic unified model, referenced as the SaaS Complexity Framework Unified Model (SCFUM). This model allows a comprehensive understanding of the common business processes and functionalities. Mapping the organisation's business structure enables the calculation of the complexity for that business, besides other analysis possibilities. The basis of this unified business model was derived through the review of the common elements of off-the-shelf ERP systems, together with business case reviews. The SCFUM model was defined through the following:

- Extraction of the functional elements comprising each ERP system.
- Investigation of the functional elements and interaction within the wider processes.
- Compilation of common functional elements and related processes across systems within the unified model.

These elements were then modelled as follows:

1. The **Process Functional Elements** – these being the lowest level of detail representing the process functional elements of the business data entities.
2. The **Process Nodes** – these being the general processes made up as a collation of process functional elements, forming a business module or functional group.

10.2.3 The Derivation of Business Complexity for 9 Business Case Studies

This work validated the SCF framework through its application to 9 business case studies, (real enterprises with anonymised datasets), detailed in Chapter 7. Each business case represented a factual complex business example and applied an anonymised business dataset, extracted from each of the business's current systems, detailed in Table 7. Using the SCF framework, these business cases provided the means to:

- Detail step by step the SCF framework application process in a real business scenario.
- Examine the data preparation and normalisation process prior to mapping to the SCFUM model.
- Review variances within the mapping process between the business cases and the SCFUM model, enabling the calibration of the model.
- Calculate the business complexity in units of SCI, for each business case.
- Calculate the business complexity in units of SCI, for off-the-shelf ERP solutions (AX and NAV).

10.2.4 Analyses and Interpretation of the Business Case Results

The results derived from the business cases enabled the analysis of each of these respective businesses from multiple facets and perspectives. This element of the research, detailed in Chapter 8, enabled the following outcomes:

- Analysis in detail of the resultant data, enabling the interpretation of these for each business and their common industry sector.
- Evaluating the SCI minimum and maximum complexity requirements for each business case, in relation to the test ERP solutions.
- Deriving various business intelligence data in support of a possible strategy or deployment projects.

10.2.5 A Strategy for Service Provider Transition to Cloud Hosted Model

Chapter 9 discussed how the service provider could explore the solution design considerations to develop a cloud transition strategy. Expanding on the results derived from the business case application of the SCF framework, various strategy options have been proposed and analysed, both individually and holistically. The risk and investment return elements for each strategy was evaluated, and considered within the wider commercial context of the possible target markets identified. A summary proposal was then presented, that considered the commonalities within the various strategic options. The author concluded that a holistic cloud transition strategy, that would cover the five options, would reduce transition risk and cost, whilst provided a cloud presence within a shorter time-frame. These strategies will need to be validated in practice through future work.

10.2.6 A Cloud Systems Architecture Proposal for Cloud ERP

Section 9.7 extrapolated on the evolution of ERP and systems architecture, proposing that for a complex business system to be better adapted to a SaaS model, complexity needs be disaggregated in smaller and more manageable components within the architecture. These could then be brought back together again to form the whole solution. Having web services tracking the SCF process functional elements would allow different parts of the solution to be potentially provided by one or more integrated service providers, and allowing for other parts of the solution to be retained on-premises, as a hybrid deployment model.

Over the progress of the research, the validity of this approach was evidenced by the direction the main industry players have taken, as the multi-tiered model continued to evolve in complexity, though conceptually remaining the same. Many release iterations of traditional ERP systems have shown a continuous restructuring of the underlying solution architecture, enabling increasing levels of cloud integration and support.

10.3 Future Applications of the Framework

The research advances work on the issues in the application of SaaS to complex business systems mapped to exemplars of ERP, investigating the functional elements within an implementation. Within this context, the research provides a framework around the adoption of the SaaS model for complex business. This provides organisations with a myriad of practical applications for this innovative framework within the decision-making process. As shown in Figure 19, examples of such business applications could be as follows:

- **Risk assessments** – to apply a scientific approach at quantitatively evaluating risk for a SaaS implementation, using SCFUM model and SCI metrics. This could be applied by both business users and service providers. This framework could also be applied to traditional deployment models.

- **Cost and Budget forecasts** – through the calculation of SCI metric and the integration of this against published pricing models for SaaS services, a business could derive possible implementation and usage costs for budgeting purposes.
- **SaaS service applicability** – To research and quantify internal business complexity, allowing this to be used as a measure for mapping business process to existing and future SaaS ERP solutions, identifying applicability and shortfalls.
- **Industry / Sector Analysis for R&D investment** – to further investigate a service provider business solution complexity, as a comparison of average industry complexity through the calculation of respective SCIs. This measure would provide a view of R&D investment scale requirements.
- **Requirements Analysis** – Supports the analysis of a business for its processes and functions, providing a means to model these within the framework. This research noted that whilst the framework was developed for use within cloud deployments, the resultant SCF framework could also apply to traditional ERP systems and deployment models.
- **System/Service Comparison and Evaluation -** Possible automation of comparison across SaaS service providers at the process node level of detail. This could be applied in a similar fashion as to how consumers compare utility services for example.
- **Evaluation and forecasting** of cost and deployment times.
- **Explore global transition -** use this framework to model offshoring business processes, or evaluate international business culture and global process.
- **ERP testing -** to provide a framework for comparative testing of ERP vendor solutions either as an international standard such as ISO, or as an informal web service.

These applications would further support key business decisions, whilst minimising the risk and cost of implementation for both the business service consumer and service provider.

10.4 Limitations

Though the research has made several contributions to facilitate the transition of a business and its systems to cloud hosted services, several constraints are accepted and remain unresolved.

- **English as the main business language**. As such, the SCFUM model applied English syntax for its Process Nodes and Process Elements. Application of the framework within non-English systems would require an equivalent SCFUM model in that language, or a translation layer to the actual model. This applies to both language descriptors and digital character sets.

- **External processes within business complexity**. This research surmises that business complexity would increase further if disparate systems, technologies and devices were used to handle the various business elements. These would traditionally require data interfaces, mapping structures and communication protocols, these being potentially provided as services through third party services. The move away from internal system silos to multitenancy on the cloud exacerbates this issue. Whilst the SCFUM model supports such elements at a high-level, extension of the model would be required to enable separate analysis of these within the framework. Moreover, an element of complexity weighting could be applied to reflect this complexity overhead within SCI calculations.

- **Terminology variances across organisations and business systems.** A consideration of this research was that different organisations and their supporting systems are likely to have developed their own terminology for describing similar business processes and functions. These differences in terms and labels are similarly evident when comparing processes across industries. The SCFUM model was developed using generic and common business terminology to mitigate this, however application of the framework would still require a level of ontological mapping.

10.5 Future Considerations

The SCF framework is ready to be deployed beyond the laboratory. Future considerations would be required to further extend the model and to facilitate its take-up within the wider business community, for the services described in Section 10.3. The following are general future considerations to this aim:

Automation of Mapping and Calculations

The use of the SCF framework within a business environment currently requires several manual steps and calculations in mapping the business and deriving its SCI business complexity. Additional extensions would be required to facilitate this process, and automate the calculation and reporting of the framework results and recommendations. This could be enabled using the E-Commerce Extensible Mark-up Language (ECXML), which was developed through earlier work by the author (Spiteri, 2004).

Facilitate Business Analysis by the Service Consumer

Closely aligned to the automation consideration, the resultant data derived from the application of the framework should be easily accessible for analysis and presentation, ideally with graphical representation, as in Figure 32 and Figure 33. This would enable actionable information to help corporate executives, business managers and other end users make best use of the framework for more informed business decisions.

Weighted SCF Complexity Calculations

This research infers a singular relation between the complexity of the organisation and the number of interrelated process elements that contribute its business in the development of the SCI metric. This implied that each Process Element identified within the business was attributed an SCI complexity unit per element. The author recommends investigating further in extending the framework to factor various complexity weightings within the complexity calculations, probability correlated to industry sector. An example would be using the volume of data processed against an SCF process element as a complexity multiplier for that element.

Scalability

Scalability within the context of this framework is the ability to expand upon the SCFUM model as it is applied across wider and more specific industries, possibly encountering new processes. This suggests the need of a management body for the framework, that incorporates new process nodes and elements encountered in industry, into new SCFUM model revisions. This would ensure the framework remains relevant as industries change over time.

10.6 Concluding Thoughts

Understanding a business and its underlying complexity is essential in the transition of its supporting systems to technology, particularly cloud hosted services. Based on a comprehensive analysis of traditional ERP systems and business case studies, the thesis and research defined the criteria to model a business as a collation of process nodes and elements level, as well as a metric to quantify business complexity as a unit of SCI. These elements provided the foundation in the design of a new SCF framework. The author then evaluated and calibrated the framework using business case studies as a proof of concept. Example analyses and interpretation of the resultant output data were highlighted. The author is convinced that this work advances the current understanding in modelling business to cloud SaaS solutions, and believes the proposed framework can substantially facilitate the decision-making process within various technology strategies.

LIST OF ACRONYMS

API	Application Programming Interface
ASP	Application Service Provider
AWS	Amazon Web Services
AX	Microsoft Dynamics Axapta
B	Bytes
BI	Business Intelligence
BPaaS	Business Process as a Service
BSNE	Business Specific N/A Entities
CaaS	Communication as a Service
CAGR	Compound Annual Growth Rate
CE	Case Entities
CLI	Command Line Interface
CPU	Central Processing Unit
DNS	Domain Name Server
EC2	Amazon Elastic Compute
ECXML	E-Commerce Extensible Mark-up Language
ERP	Enterprise Resource Planning
GCE	Google Compute Engine
GUI	Graphical User Interface
HR	Human Resource Management
HTTP	hypertext transfer protocol
I/O	Input / Output
IaaS	Infrastructure as a Service
ISE	Industry Specific Entities
IT	Information Technology
KB	Kilobytes
KPI	Key Performance Indicators
MCE	Merged Case Entities
Mgmt.	Management
MRP	Manufacturing Requirements Planning
NA, N/A	No Value Available
NAS	Network Attached Storage
NAV	Microsoft Dynamics Navision
NIC	Network Interface Card
P2P	Peer to Peer
PaaS	Platform as a Service
PC	Personal Computer
PE	Process Element
PPC	Pay Per Click
Q&A	Questions and Answers
QC	Quality Control

QoS	Quality of Service
R&D	Research and Development
RAM	Random Access Memory
RFP	Request For Proposal
RFI	Request For Information
ROI	Return on Investment
SaaS	Software as a Service
SAN	Storage Area Networks
SCF	Spiteri (SaaS) Complexity Framework
SCFUM	Spiteri (SaaS) Complexity Framework Unified Model
SCI	Spiteri (SaaS) Complexity Index
SCM	Spiteri (SaaS) Complexity Metric
SLA	Service Level Agreements
SMB	Small and Medium sized Business
SME	Small and Medium sized Enterprise
UI	User Interface
VM	Virtual Machine
WWW	World Wide Web

LIST OF RELATED PUBLICATIONS

Spiteri, K.J., Reynolds, T., Luca, C., Wilson G., (2012a). Developing a Framework for Modelling Complex Business Systems within the Cloud. *International Conference of Information Society (i-Society 2012)*. IEEE, pp. 435 – 440

Spiteri, K.J., Reynolds, T., Luca, C., Wilson G., (2012b). Defining a Baseline Complexity Model for ERP Systems. *Journal of Internet Technology and Secured Transactions (JITST)*. Infonomics Society. Volume 1, Issue 3, pp. 86 - 92

Spiteri, K.J., Reynolds, T., Luca, C., Wilson G., (2014). Complexity Metric for Device Integration within Business SaaS. *International Conference on Optimization of Electrical and Electronic Equipment (OPTIM 2014)*. IEEE. pp. 936 - 941

Spiteri, K.J., Reynolds, T., Luca, C., Wilson G., (2015). Methodology for Deriving a Business Complexity Index through the Application of the Complexity Framework for SaaS. *13th International Conference on Industrial Informatics (INDIN 2015)*. IEEE. pp. 470 - 475

REFERENCES AND BIBLIOGRAPHY

Alpern, P. (2010). Drawing the Lines on System Integration, *Industry Week*, Vol. 259 Issue 7, pp. 49-49.

Amaral, L.A.N. and Uzzi, B. (2007). Complex Systems—A New Paradigm for the Integrative Study of Management, Physical, and Technological Systems. *Management Science* Vol. 53, No. 7, July 2007, pp. 1033–1035.

AmazonWebServices, (2017). *What Is Amazon EC2? - Amazon Elastic Compute Cloud.* [online] Amazon. Available at: http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html [Accessed Feb 2017].

Amrhein, D., Quint, S., (2009). *Cloud Computing for the Enterprise.* [online] IBM DeveloperWorks, Available at: www.ibm.com/developerworks/websphere/techjournal [Accessed June 2009].

Andrews, J. (2014). *The Subscription Pricing Journey*. [online] Zuora, Available at: https://www.zuora.com/guides/innovative-subscription-pricing-strategies/ [Accessed Oct 2017].

ARG, (2011). *QAD On Demand gives manufacturers the tools they need to become global.* [online]. Aberdeen Research Group, TheStreet.com Inc. Available at: http://bx.businessweek.com/enterprise-software/view?url=http%3A%2F%2Fc.moreover.com%2Fclick%2Fhere.pl%3Fr3936360089%26f%3D9791 [Accessed 8 July 2011].

Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R.H., Konwinski, A., Lee, G., Patterson, D.A., Rabkin, A., Stoica, I., Zaharia, M. (2009). *Above the clouds: A Berkeley view of cloud computing.* [pdf] UC Berkeley Available at: https://www2.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.pdf [Accessed June 2012].

Behringer, M.H. (2009). Classifying Network Complexity, ACM ReArch'09 Workshop

Betke, B. (2010). *ERP Software in the Multichannel World.* [online] F. Curtis Barry and Company, Available at: http://www.fcbco.com/articles-whitepapers/erp-software-multichannel-world.asp [Accessed Oct 2010].

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

Bukisa.com, (2009). *The History of Networking.* [online] Available at: http://www.bukisa.com/articles/144193_the-history-of-networking [Accessed June 2015].

BusinessDictionary. (2015). *What is distributed computing (DC)? definition and meaning.* [online] BusinessDictionary.com Available at: http://www.businessdictionary.com/definition/distributed-computing-DC.html [Accessed June 2015].

Buyya, R., Broberg, J., Goscinski, A. (2011). *Cloud Computing: Principles and Paradigms.* Hoboken: John Wiley & Sons.

Buyya, R., Pandey, S., Vecchiola, C. (2009). Cloudbus toolkit for market-oriented cloud computing. *1st International Conference on Cloud Computing (CLOUDCOM 09),* Berlin: Springer. pp. 24–44.

Campbell, A.T., Coulson, G., Kounaviset, M.E. (1999). Managing Complexity Middleware Explained. *IT Professional*, IEEE. Vol.1, pp.23-28.

Card, D.N. and Agresti, W.W. (1988). Measuring Software Design Complexity. *The Journal of Systems and Software,* (8)3, pp. 185-197.

Carr, J. (2017). *Diminishing Returns from Legacy Technology?* [online] Industry Perspectives, Available at: http://www.datacenterknowledge.com/industry-perspectives/diminishing-returns-legacy-technology [Accessed Sep 2017].

Carraro, G. (2006). *SaaS: Architecture Strategies for Catching the Long Tail.* [online] Microsoft Knowledge Base, Available at: www.microsoft.com/serviceproviders/saas/default.mspx [Accessed Jun 2009].

Ceruzzi, P.E. (2003). A History of Modern Computing. *MIT Press,* pp. 220-221

Chang, She-I. (2010). Critical Factors of ERP Adoption for Small- and Medium- Sized Enterprises: An Empirical Study, *Journal of Global Information Management,* Vol. 18 Issue 3, pp. 82-106.

Chappell, D. (2008), *A Short Introduction to Cloud Platforms.* [pdf] David Chappell & Associates, Available at: www.davidchappell.com/CloudPlatforms--Chappell.pdf, [Accessed May 2009]

Checkland, P. (1981). Systems Thinking, Systems Practice, Hoboken: John Wiley & Sons, pg. 78.

Claessens, M. (2017). *Product Life Cycle Stages (PLC) – Managing the Product Life Cycle.* [online] Marketing-Insider, Available at: https://marketing-insider.eu/marketing-explained/part-iii-designing-a-customer-driven-marketing-strategy-and-mix/product-life-cycle-stages/ [Accessed Oct 2017]

Coffman, K. and Odlyzko, A. M. (1998). *The size and growth rate of the Internet.* [pdf] AT&T Labs. Available at: http://www.dtc.umn.edu/~odlyzko/doc/internet.size.pdf [Accessed June 2011].

Colombus, L. (2011). *Enterprise Resource Planning.* [online] Quora, Available at: <www.quora.com/Louis-Columbus/Enterprise-Resource-Planning-1/answers> [Accessed Sep 2011].

CSCC, (2015). *Practical Guide to Platform -as-a-Service* [pdf] Cloud-Council. Available at: http://www.cloud-council.org/CSCC-Practical-Guide-to-PaaS.pdf [Accessed Jun 2015].

Dabrowski, C. (2009). Reliability in grid computing Systems. *Concurrency and Computation: Practice and Experience.* [pdf] Wiley InterScience, Available at: https://www.nist.gov/sites/default/files/documents/itl/antd/Dabrowski-GridReliabilityEarlyView.pdf [Accessed June 2011].

Defelice, A. (2010). Cloud computing: what accountants need to know, *Journal of Accountancy*, v210 i4, pp.50-56.

Defelice, A. (2010). Technology is a cornerstone of Successful CPA Firms, *Journal of Accountancy*. [online] Available at: <http://www.journalofaccountancy.com/Issues/2010/Jan/20092309.htm> [Accessed Oct 2010].

Edmonds, B. (1999). “A Definition of Complexity”, Syntactic Measures of Complexity, University of Manchester, Chapter 4

Epstein, M.J., Buhovac, A.R. (2006). What’s in IT for you (and your company)?, *Journal of Accountancy*, 2006. [online] Available at: http://www.journalofaccountancy.com/Issues/2006/Apr/WhatSInItForYouAndYourCompany.htm [Accessed Oct 2010].

Exforsys, (2007). *Client Server Architecture.* [online] Exforsys.com, Available at: http://www.exforsys.com/tutorials/client-server/client-server-architecture.html [Accessed June 2011].

Falgout, J. (2011). Dataflow Programming: Handling Huge Data Loads Without Adding Complexity. [online] Available at: http://www.drdobbs.com [Accessed June 2012].

Fenton, N.E. and Pfleeger, S.L. (1997). *Software Metrics: A Rigorous and Practical Approach. 2nd edn.* Boston: International Thomson Publishing.

Forbes, (2015). *Roundup Of Cloud Computing Forecasts And Market Estimates Q3 Update, 2015.* [online] Forbes.com. Available at: https://www.forbes.com/sites/louiscolumbus/2015/09/27/roundup-of-cloud-computing-forecasts-and-market-estimates-q3-update-2015/#71e7a0ad2308 [Accessed June 2015].

Force.com, (2009). *A Comprehensive Look at the World’s Premier Cloud-Computing Platform,* [pdf] Salesforce white paper, Available at: http://www.developerforce.com/media/Forcedotcom_Whitepaper/WP_Forcedotcom-InDepth_040709_WEB.pdf, [Accessed Jun 2011].

Forrester research, (2011), *Sizing The Cloud – A BT Futures Report Understanding And Quantifying The Future Of Cloud Computing.* [online] Forrester.com, Available at: http://www.forrester.com/Sizing+The+Cloud/ [Accessed Nov 2011].

Furht, B. and Escalante, A. (2010). *Handbook of Cloud Computing.* New York: Springer Science.

Gartner, (2014). *Microsoft AX Magic Quadrant for Single¬Instance ERP for Product¬Centric Midmarket Companies.* [online] Garnter.com. Available at: https://www.gartner.com/doc/2926417/magic-quadrant-singleinstance-erp-productcentric [Accessed June 2015].

Gartner, (2015). *Modernization and Digital Transformation Projects Are Behind Growth in Enterprise Application Software Market.* [online] Gartner.com. Available at: http://www.gartner.com/newsroom/id/3119717 [Accessed June 2016].

Gartner, (2016). *By 2020, a Corporate "No-Cloud" Policy Will Be as Rare as a "No-Internet" Policy Is Today* [online] Gartner.com. Available at: http://www.gartner.com/newsroom/id/3354117 [Accessed June 2016].

Google, (2016). *Google Could Platform.* [online] Google.com. Available at: https://cloud.google.com/compute/docs/faq [Accessed Nov 2016].

Goral, T. (2009). 10 questions and answers about the cloud: what you need to know now about the next big thing in internet technology. *University Business*, Professional Media Group LLC, vol.12 i3, pp.47(2).

Hall, K. (2010), *Gartner: SaaS sales will grow 16.2% to $10.7bn in 2011.* [online] ComputerWeekly.com. Reed Business Information Ltd., Available at: http://www.computerweekly.com/Articles/2010/12/14/244489/Gartner-SaaS-sales-will-grow-16.2-to-10.7bn-in-2011.htm. [Accessed July 2011].

Hamdaqa, M. (2012). Cloud Computing Uncovered: A Research Landscape. *Advances in Computers.* Elsevier Press. Vol.86, pp. 41-85.

Hart J., (2007), *Fundamentals of Enterprise Resource Planning ERP*. [online] eZine Articles, Available at: <http://ezinearticles.com/?Fundamentals-of-Enterprise-Resource-Planning-ERP&id=787263 > [Accessed Oct 2010].

Henry, S., (1981), *Software Metrics Based on Information Flow*, [journal] IEEE Trans. Software Eng. 7, 510-518.

Henry, S., and Kafura, D., (1984), *The Evaluation of Systems' Structure Using Quantitative Metrics*, Software Pratt. Exp. 14, 561-573.

Hill, D., Meier, J.D., Homer, A., Taylor, J., Bansode, P., Wall, L., Boucher, R., Bogawat, A. (2009). *Microsoft Application Architecture Guide.* [online] Microsoft.com. Available at: https://msdn.microsoft.com/en-us/library/ee658086.aspx [Accessed June 2012].

HIPAA, (2009). What is cloud computing? Name is symbolic, services are real, *Briefings on HIPAA*, Vol.9, i12, pp. 10-11.

Hoppermann J., (2010). The Definition of Complexity Is A Complex Matter. [online] Forrester blogs, Available at: <http://blogs.forrester.com/jost_hoppermann/10-04-15-definition_complexity_complex_matter> [Accessed June 2011].

Hvolby, H.H. and Trienekens, J.H. (2010). Challenges in Business System Integration. *Computers in Industry*, 61 (9): pp. 808-814.

IBM Archives, (2015). *History of computing hardware (1960s-present).* [online] IBM Archives. Available at: https://www-03.ibm.com/ibm/history/ [Accessed June 2015].

IBM, (2016). *IBM Cloud - Cloud Computing for Builders & Innovators* [online] ibm.com. Available at: https://www.ibm.com/cloud-computing/ [Accessed Nov 2016].

Internet World Stats, (2017). *Internet Usage and Population Statistics.* [online] Miniwatts Marketing Group, Available at: http://www.internetworldstats.com/stats.htm [Accessed Feb 2017].

ITSM Portal, (2010). *The concept of vendor lock-in and how it relates to cloud computing.* [online] CA Community, Available at: https://www.ca.com/en/blog-highlight/the-concept-of-vendor-lock-in-and-how-it-relates-to-cloud-computing.html [Accessed Oct 2017].

Jagersma, P.K. (2004). *Managing Business Complexity*. [online] ManagementSite, Available at http://www.managementsite.com/461/Managing-Business-Complexity.aspx [Accessed Sept 2011].

Jones, T. (2009). *IBM Cloud computing with Linux.* [online] ibm.com. Available at: http://www.ibm.com/developerworks/library/l-cloud-computing/ [accessed June 2012].

Kahn, R.E. and Cerf, V.G. (1999). *What Is the Internet (And What Makes It Work).* [online] CNRI. Available at: http://www.cnri.reston.va.us/what_is_internet.html [Accessed June 2011].

Kearney, J.K., Sedlmeyer, R.L., Thompson, W.B., Gray, M.A., Adler, M.A. (1986). Software Complexity Measurement. *Communications of the ACM*, Vol. 29 No.11.

Kimberling, E. (2008). *SaaS vs. Traditional ERP: Five Key Differentiators.* [online] Panorama Consulting Group, Available at: http://panorama-consulting.com/saas-vs-traditional-erp-five-key-differentiators/, [Accessed Sep 2010].

Kimberling, E. (2010). *SaaS, On Premise, Cloud, Best of Breed: Making Sense of All the ERP System Options.* [online] Panorama Consulting Group, Available at: http://panorama-consulting.com/saas-on-premise-cloud-best-of-breed-making-sense-of-all-the-erp-system-options/, [Accessed Oct 2010].

Kushida, K.E., Murray, J., Zysman, J. (2011). Diffusing the Cloud: Cloud Computing and Implications for Public Policy. *Journal of Industry, Competition and Trade.* [online] Springerlink.com, Vol. 11, Issue 3, pp 209–237.

Latva-Koivisto, A.M. (2001). *Finding a complexity measure for business process models.* [pdf] Helsinki University of Technology, Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.25.2991&rep=rep1&type=pdf [Accessed June 2011].

Leiner, B.M., Cerf, V.G., Clark, D.D., Kahn, R.E., Kleinrock, L., Lynch, D.C., Postel, J., Roberts, L.G., Wolff, S. (2012). *Brief History of the Internet.* [online] Internet Society. Available at: http://www.internetsociety.org/internet/what-internet/history-internet/brief-history-internet [Accessed June 2013].

Leon, A. (2014). *ERP Demystified, 3rd edn.* India: McGraw Hill Publishers.

Leon, J.F. (2010). Vetting a Vendor: Questions to Ask Before Making an Investment, *Journal of Accountancy.* [online] Available at: http://www.journalofaccountancy.com/Issues/2010/Oct/VettingaVendor.htm, [Accessed Sep 2010].

Levy, D.L. (2000). *Applications and Limitations of Complexity Theory in Organization Theory and Strategy.* [pdf] University of Massachusetts. Available at: http://www.faculty.umb.edu/david_levy/complex00.pdf [Accessed Oct 2010].

Lucas, C. (2006). *Quantifying Complexity Theory* [online] Available at: http://www.calresco.org/lucas/quantify.htm, [Accessed Sept 2011].

Mabert, V.A., Soni, A., Venkataramanan, M.A., Ewing, M. (2001). Enterprise Resource Planning: Common Myths vs. Evolving Reality, *Harvard Business Review*. Harvard Business Publishers, pp. 1-8.

Marston, T. (2012). *What is the 3-Tier Architecture?* [online] Available at: http://www.tonymarston.co.uk/php-mysql/3-tier-architecture.html [Accessed June 2013].

Martin, J.F., (2009). *Building SaaS* [online] IQNavigator, Available at: http://buildingsaas.typepad.com/blog/business_model/index.html [Accessed Aug 2010].

Mastorakis, G., Mavromoustakis, C.X., Pallis, E. (2014). *Resource Management of Mobile Cloud Computing Networks and Environments.* Berlin: Springer.

McCabe, T.J. (1976). A complexity measure, Software Engineering, *IEEE Transactions*, SE-2, no. 4, pp. 308–320.

Megiddo, N. (1987). On the complexity of linear programming. *Advances in economic theory: Fifth world congress,* Cambridge University Press, Cambridge, pp. 225–268.

Mell, P. and Grance, T. (2011). *The NIST definition of cloud computing.* [pdf] NIST. Available at: http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf [Accessed June 2012].

Metcalfe, R. (1996). *Packet Communication.* California: Peer-to-Peer Communications Inc.

Microsoft, (2006). *Software as a Service (SaaS): An Enterprise Perspective.* [online] Microsoft.com. Available at: https://msdn.microsoft.com/en-us/library/aa905332.aspx [Accessed June 2012].

Microsoft, (2016). *Microsoft Azure: Cloud Computing Platform & Services.* [online] Microsoft.com. Available at: https://azure.microsoft.com/en-gb/ [Accessed Nov 2016].

Mikulecky, D.C. (2001). Definition of Complexity, *Virginia Commonwealth University*, [online] Available at: http://views.vce.edu/ mikuleck/ [Accessed Nov 2012].

Miniwatts Marketing Group, (2017). *Internet World Stats*. [online] MMG. Available at: http://www.internetworldstats.com/ [Accessed March 2017].

Mitchell, B. (2016). *Introduction to Client Server Networks*. [online] Lifewire.com. Available at: https://www.lifewire.com/introduction-to-client-server-networks-817420 [Accessed Nov 2016].

Moody, D.L. and Flitman, A. (2000). A Decomposition Method for Entity Relationship Models, *1st International conference on systems thinking in management* [pdf] Available at: https://pdfs.semanticscholar.org/90ec/420f0a70b77a155182af863a3faf647211a1.pdf [Accessed Nov 2010].

Motwani, J., Subramanian, R., Gopalakrishna, P. (2005). Critical factors for successful ERP implementation: Exploratory findings from four case studies. *Computers in Industry*, Vol.56, Issue 6, pp. 529-544.

Muketha, G.M., Ghani, A.A.A, Selamat, M.H., Atan, R. (2010). a Survey of Business complexity metrics. *Information Technology Journal*, pp. 1336-1334.

Mulcahy, R. (2007). Business Intelligence Definition and Solutions. [online] CIO.com, Available at: http://www.cio.com/article/2439504/business-intelligence/business-intelligence-definition-and solutions.html [Accessed Jan 2017].

ND Liyanage, (2013). *Client/Server Architecture. Advantages and Disadvantages*. [online] clientserverarch.com Available at: http://clientserverarch.blogspot.co.uk/ [accessed June 2015].

Neto, M.D. (2014). *A brief history of cloud computing*. [online] IBM Archive. Available at: https://www.ibm.com/blogs/cloud-computing/2014/03/a-brief-history-of-cloud-computing-3/ [Accessed June 2015].

Nicolis, G. and Nicolis, C. (2009). Foundations of Complex Systems. *European Review*, Vol. 17, Issue 02, pp. 237-248.

Osintsev, A. (2016). *Tier 1 vs. Tier 2 vs. Tier 3 ERP Systems: What's the Difference, Anyway?* [online] TEC, Available at: https://www3.technologyevaluation.com/research/article/tier-1-vs-tier-2-vs-tier-3-erp-whats-the-difference-anyway.html [Accessed Nov 2016].

Panorama Consulting, (2010). *Software as a Service (SaaS) Selection*. [online] Panorama Consulting Group Available at: <http://panorama-consulting.com/services/erp-software-selection/ > [Accessed Oct 2010].

Panorama Consulting, (2016). *Clash of the Titans 2016: An Independent Comparison of SAP, Oracle and Microsoft Dynamics*. [online] Panorama Consulting, Available at: http://go.panorama-consulting.com/rs/603-UJX-107/images/Clash-of-the-Titans-2016.pdf [Accessed Nov 2016].

Parallels, (2014). *Parallels SMB Cloud Insights reports, 2014*. [pdf] Odin.com. Available at: http://www.odin.com/fileadmin/parallels/documents/smb-reports/2014/Parallels_SMB_report_UK_20140930_web.pdf [Accessed June 2016].

Parallels, (2015). *Parallels SMB Cloud Insights reports, 2015*. [pdf] Odin.com. Available at: http://www.odin.com/fileadmin/parallels/documents/smb-reports/2015/SMB_Global_EN_web_20150602.pdf [Accessed June 2016].

Patni, J.C., Aswal, M.S. (2011). Load balancing strategies for Grid computing. *Electronics Computer Technology (ICECT)*, IEEE. Vol 3, pp.239-243.

Perrone, W.A., Heaphy, M.W., Mahajan, S.D., (2010). *Cloud Computing: Why Forecast Should Matter To You.* [online] Connecticut Law Tribune, Available at: http://www.ctlawtribune.com/getarticle.aspx?ID=38520, [Accessed Oct 2010].

Rackspace, (2016). *OpenStack Private Cloud | Fully-Managed OpenStack Solutions.* [online] Rackspace.com. Available at: https://www.rackspace.com/en-gb/openstack/private [Accessed Nov 2016].

Rashid, M.A., Hossainet, L., Patrick, J.D. (2002). *The Evolution of ERP Systems: A Historical Perspective.* Oakland: Idea Group Publishing, pp. 1-16.

Reimer, J. (2005). *"Personal Computer Market Share: 1975–2004".* [online] Ars Technica. Available at: http://www.jeremyreimer.com/total_share.html [Accessed 2008-02-13].

Richardson, K.A. (2004). Systems Theory and Complexity, *Emergent Publications,* Issue Vol. 6 No. 4 pp. 77-82.

Rouse, M. (2008). *Client/Server Model, Client/Server Architecture.* [online] TechTarget. Available at: http://searchnetworking.techtarget.com/definition/client-server [Accessed June 2011].

Rouse, M. (2008). *TCP/IP (Transmission Control Protocol/Internet Protocol)-Definition.* [online] WhatIs.com. Available at: http://searchnetworking.techtarget.com/definition/TCP [Accessed June 2011].

Rouse, M. (2008). *What is Master/Slave? - Definition* [online] TechTarget. Available at: http://searchnetworking.techtarget.com/definition/master-slave [Accessed June 2011].

Rouse, M. (2015). *Distributed Computing-Definition*. [online] WhatIs.com. Available at: http://whatis.techtarget.com/definition/distributed-computing [Accessed Sept 2015].

Rouse, M. and O'Reilly, J. (2015). *Definition of Grid Computing.* [online] WhatIs.com. Available at: http://searchdatacenter.techtarget.com/definition/grid-computing [Accessed Sept 2015].

Sarrell, M. (2010*). Interest Growing in Private Cloud Computing.* [online] eWeek, Ziff Davis Enterprise, Available at: <http://www.eweek.com/c/a/Cloud-Computing/Interest-Growing-in-Private-Cloud-Computing-444314/ > [Accessed Oct 2010].

Schneier, B. (2009). *Cloud Computing.* [online] The Guardian, Available at: <www.schneier.com/blog/archives/2009/06/cloud_computing.html> [Accessed Oct 2010].

Shapiro, F.R. (2000). "Annals of the History of Computing". IEEE Annals of the History of Computing. *IEEE Journal.* 22 (4): pp. 70–71.

Shawish, A. and Salama, M. (2014). Cloud Computing: Paradigms and Technologies. *Inter-cooperative Collective Intelligence: Techniques and Applications.* Berlin: Springer.

Shin, I. (2006). Adoption of Enterprise Application Software and Firm Performance, *Small Business Economics*, 26(3), pg. 241.

Simon, H.A. (1962). The architecture of complexity: Hierarchic systems. *Proceedings of the American Philosophical Society,* 106(6), pp. 467–482.

Simon, H.A. (1973). The Organisation of Complex Systems. *Hierarchy Theory - The Challenge of Complex Systems,* New York: Braziller.

Singh, P. (2005). *The Practical Handbook of Internet Computing.* Florida: CRC Press LLC. pg. 803.

Skyttner, L. (2001), *General Systems Theory: Ideas and Applications*, NJ: World Scientific.

Snowden, M. (2010). *The CEO's Guide to Cloud Computing.* Auckland: OneNet. Pg.18.

Spiteri, K.J. (2004). *Regulating E-Commerce through Extensible Mark-up Languages.* [MSc. Thesis] University of Malta.

Tarzey, B. (2008). *The truth about software as a service.* [online] Quocirca, Available at: software.silicon.com/applications/0,39024653,39169781,00.htm [Accessed Aug 2010].

Tassabehji, R. (2005). *Applying E-Commerce in Business.* London: Sage Publications. 2nd ed. pg. 43.

Tech-Faq, (2009). *What is ERP?* [online] Topbits Tech Community, Available at: http://www.tech-faq.com/erp.shtml [Accessed Sep 2010].

Technopedia, (2015). *What is Network Computing? – Definition.* [online] Technopedia. Available at: https://www.techopedia.com/definition/23619/network-computing [Accessed June 2015].

Vass, B. (2009). *Delivering on the Promise of Software as a Service (SaaS)* [online] Sun.com, Available at: http://www.sun.com/emrkt/innercircle/newsletter/0906edchoice.html [Accessed Feb 2011].

Voas, J. and Zhang, J. (2009). Cloud Computing: New wine or just a new bottle? *IEEE ITPro*, pp. 15-17.

Wailgum, T. (2010). *ERP (Enterprise Resource Planning) topics covering definition, objectives, systems and solutions*. [online] CIO.com. Available at: http://www.cio.com/article/40323/ERP_Definition_and_Solutions [Accessed Sept 2011].

Web Based ERP, (2006). *Web Based ERP Software.* [online] Available at: www.web-based-erp-software.com/ [Accessed Sep 2010].

Welch, J. and Kordysh, D. (2007). *Seven Keys to ERP Success.*, Strategic Finance, 89(3), pp. 40-47, 61.

Wildstrom, S.H. (2006). What to entrust to the Cloud. *Business Week*, Mochila Inc, i4125, pg. 89.

Wilson Perumal & Company, (2011). *Waging war on complexity in supply chains*. [pdf] Available at: 150213apicshouston-wagingwaroncomplexityinsupplychainsvf-150215143627-conversion-gate01.pdf

Wittmann, A. (2010). What Cloud Computing Really Means.: Practical Analysis, *InformationWeek*, i1263, pg. 40.

Wolfram, S. (1985). *Complex Systems Theory*. [Tech.Rep.] Institute for Advanced Study, Princeton.

Woodford, C. (2016). *Computer Networks.* [online] explainthatstuff.com. Available at: http://www.explainthatstuff.com/howcomputernetworkswork.html [Accessed June 2015].

World Heritage Encyclopedia, (2015). *Middleware (distributed applications).* [online] Available at: http://www.gutenberg.us/articles/middleware_(distributed_applications) [Accessed June 2016].

Ye, N., Zhu, Y., Wang, R., Malekian, R., Qiaomin, L. (2014). An Efficient Authentication and Access Control Scheme for Perception Layer of Internet of Things, *Applied Mathematics and Information Sciences*, Vol. 8, No. 4, pp. 1617-1624.

Young, B., Booch, G., Maksimchuk, R.A., Engle, M.W., Conallen, J., Houston, K.A. (2007). Organised and Disorganised Complexity, *Object-Oriented Analysis and Design with Applications*. Boston: Addison-Wesley Professional.

Zandbergen, P., (2015). *What is a Mainframe System?* [online] Study.com. Available at: http://study.com/academy/lesson/what-is-a-mainframe-system-lesson-quiz.html [Accessed July 2017].

APPENDIX A – SCFUMV2

This appendix contains the final iteration of the SaaS Complexity Framework Unified Model (SCFUMv2).

Process Node	Business Functional Element	Description
Administration	Administration	System administration and setups
Administration	Business Integration	Management and processing of business integration and interfaces
CRM	Contacts	Management of business contacts
CRM	CRM Management	Contact Relationship Management
Finance	Bank Account Reconciliations	Processing of financial bank reconciliations
Finance	Bank Accounts	Management of business bank accounts
Finance	Budgets	Financial budgeting for expected value of revenue and expenses
Finance	Cash Receipt Journals	Processing of cash receipts into the business
Finance	Chart of Accounts	Financial management reporting
Finance	Deposits	Processing of bank deposits
Finance	Fixed Asset Journals	Processing of fixed asset related transactions
Finance	Fixed Assets Registers	Reporting on audit trail for Fixed asset transactions
Finance	Finance Mgmt	Management of Finance system
Finance	Fixed Assets	Management of business fixed assets records
Finance	General Journal	Processing of financial transactions to the general ledger
Finance	General Ledger Registers	Audit trail for financial transactions
Finance	Insurance	Management of insurances in place for Fixed assets
Finance	Insurance Journals	Processing of insurance related transactions
Finance	Intercompany	Support for intercompany transactions in a multi-company environment
Finance	Payment Journals	Processing of payments to suppliers
Finance	Tax Mgmt	Management of Tax returns
Finance	XBRL Mgmt	XBRL Business reporting
HRM	Agency Mgmt	Management of external agency labour
HRM	Benefits	Management of employee benefits
HRM	Causes of absence	Process for recording resource absence and reasons
HRM	Causes of inactivity	Process for recording resource inactivity and reasons
HRM	Disciplinary	Management of employee disciplinary records
HRM	Employee	Management of employee personal records
HRM	Employee portal	Employee reporting and data access through portal
HRM	Employment Contracts	Management of employee contracts of employment
HRM	Expenses	Management of employee expenses
HRM	HR Interfaces	Management of HR system interfaces
HRM	Human Resources	Management of the organisation's human resources for internal and statutory reasons
HRM	Interview Mgmt	Interview process management and records
HRM	Payroll	Management of employee payroll

HRM	Qualifications	Management of employee qualification records
HRM	Relatives	Management of employee records for next of kin
HRM	Time & attendance	Time and attendance records
HRM	Unions	Management of employee records for union affiliations
HRM	Workflow	Management of process workflow
Jobs	Jobs	Job / Project Management
Logistics	Bookings	Management of booking records for trips handling requests
Logistics	Courier Mgmt	Management of courier records and transactions
Logistics	Fleet Mgmt	Management of transport vehicle fleets
Logistics	Loading Slots	Management of loading / unloading driver appointment slots
Logistics	Logistics Ledger	Audit trail for transport charge transactions processing
Logistics	Route Optimisation	Route optimisation for transport planning
Logistics	Stock Load positions	Records for stock load positions on trips
Logistics	Transport Agents	Management of transport agent records
Logistics	Transport Charges	Management of transport related charges
Logistics	Transport Contract Mgmt	Management of transport contracts
Logistics	Transport Goods	Management of transport goods and products
Logistics	Transport Locations	Management of Transportation locations, hubs, depots records
Logistics	Transport Planning	Planning for transportation and trips
Logistics	Transport Requests	Management of transportation requests and requirements
Logistics	Transport Rules	Management of transport rules for handling requests
Logistics	Trips	Management of Trips, Loads, Trucks within transportation
Procurement	Consignment Mgmt	Management of consignment product reporting, traceability and profitability
Procurement	Manifest Mgmt	Management of manifest records, import of advanced shipping notices, port deliveries
Procurement	Purchase Invoices	Processing of supplier invoices for product or services receipt
Procurement	Purchase Order Archive	Archiving of different versions of purchase documents
Procurement	Purchase Orders	Purchase processing for supply replenishment
Procurement	Purchase Prices	Management of the business purchase prices for various suppliers
Procurement	Purchase Quotes	Processing and record of quotes received for required product or service
Procurement	Requisitions	Purchase requisition process for supply replenishment
Procurement	Vendors	Management of supplier records
Production	Assembly Kit Mgmt	Management and processing of assembly kits
Production	Capacity Journals	Processing of capacity transactions for Machine and work centres against production
Production	Consumption	Process for recording consumption of product within production process
Production	Finished Production Orders	Process for closing off a completed production order
Production	Firm Planned Production Orders	Process for shorter term planning production requirements for the shop floor
Production	Label Room Mgmt	Management of label printing process for product, tray-ends, pallets
Production	Machine Centres	Management of machine centres making up production routings
Production	Output	Process for recording output of product within the production process
Production	Planned Production Orders	Process for medium term planning production requirements for the shop floor
Production	Planning Worksheets	Process for managing supply and demand, and action results

		within business
Production	Production BOM	Management of bill of materials for production
Production	Production Forecasts	Volume forecasting for meeting of sales demand through production or product manufacturing
Production	Production Routings	Management of product routings for production
Production	Production Schedule	Management of the schedule for the organisation's production requirements
Production	Released Production Orders	Process for the release of production orders for the shop floor to action
Production	Resources	Management of resources to the business particularly for production
Production	Shop Calendars	Management of shop floor work calendars
Production	Touchscreen Interface	Processing of production transactions through touchscreen interface
Production	Work Centres	Management of work centres making up production routings
Production	Work Shifts	Management of different work shifts a business might have particularly in production
QC	Mobile QC	Processing of QC testing through mobile interfaces
QC	QC Archive	Archive for QC tests and records
QC	QC Results	Processing and evaluation of QC testing results, stock blocking and updates
QC	QC Schedule	Management of QC testing plans and schedules
QC	QC Tests	Management of Quality control and quality assurance testing
Sales	Blanket Sales Orders	Sales agreements with a customer in place over a period
Sales	Complaints Mgmt	Management and processing of product related complaints
Sales	EDI Interface	Management and processing of Electronic data interchange interfaces
Sales	Market Trading Mgmt	Management and processing for market trading
Sales	Promotions Mgmt	Management of production promotions and opportunities
Sales	Sales Forecasts	Management of sales forecast data within the business
Sales	Sales Invoices	Processing of customer invoices for product or services delivered
Sales	Sales Order Archive	Archiving of different versions of sales documents
Sales	Sales Orders	Sales process for product or service for generation of revenue
Sales	Sales Prices	Management of the business product sales prices, discounts etc.
Sales	Sales Quotes	Processing and submission of customer quotes in support of sale of product or service
Sales	Salespeople	Management of Sales person records
Sales	Customers	Management of customer records
Sales / Procurement	Credit Notes	Processing of credit adjustments to customer or supplier accounts
Sales / Procurement	Payment Methods	Management of payment methods for purchases and sales
Sales / Procurement	Payment Terms	Management of payment terms
Sales / Procurement	Return Orders	Product return process back to Supplier
Services	Services Mgmt	Service Management
Warehouse	Bin Contents	Management of bins and bin content within a storage location
Warehouse	Device Integration	Management and processing of device interfaces and integration
Warehouse	Inventory	Audit trail for Inventory transactions
Warehouse	Item Journals	Processing of stock counts, stock adjustments and stock movements
Warehouse	Mobile Processing	Management and processing of stock through mobile interfaces
Warehouse	Pallet Mgmt	Management and processing of pallets and pallet tracked product

Warehouse	Stock Analysis	Analysis and reporting of stock transactions and inventory
Warehouse	Stock Card	Management of stock records and product details
Warehouse	Transfer Orders	Transfer of stock from one storage location to another within the business
Warehouse	Warehouse Locations	Management of the organisation's product storage locations
Warehouse	Warehouse Movements	Processing of movement of goods within storage locations
Warehouse	Warehouse Picks	Processing of directed product picks from a storage location
Warehouse	Warehouse Put-aways	Processing of directed product put-aways within a storage location
Warehouse	Warehouse Receipts	Processing of product receipts within storage locations
Warehouse	Warehouse Shipments	Processing of product dispatch from storage locations

APPENDIX B – FUNCTIONAL REQUIREMENTS TEMPLATE

This appendix contains a "Functional Requirements Template", part of the RFP process. This template is a tool, proprietary to Plante Moran[1], created to analyse how different solution providers (Vendors) meet the requirements of a business. This enables the comparison of various vendors in the provision of the same business requirements. This is a common process undertaken when a complex business is initiating a solution implementation project. This template is presented here as an example of how traditionally such vendors are evaluated. The results from this process enable a summary of how solutions can meet requirements, as well as deriving a comparable indication of cost and risk. This information provides the business a basis on which a strategic decision for vendor and solution selection is made.

This approach poses some limitations however, since the response process is subjective in nature, and relies on the vendor understanding the requirements at face value. Based on which the level of compliance of the proposed solution is then suggested. This limitation is mitigated somewhat through the involvement of Third party advisory firms (as presented here), in helping bridge this knowledge gap, as there would be a better understanding of the potential pitfalls within the selection process. The process still relies on the subjective expertise of the advisory firm, for both the business requirements as well as the solutions under review. Additionally, whilst minimising the risk, this does incur additional costs to the business.

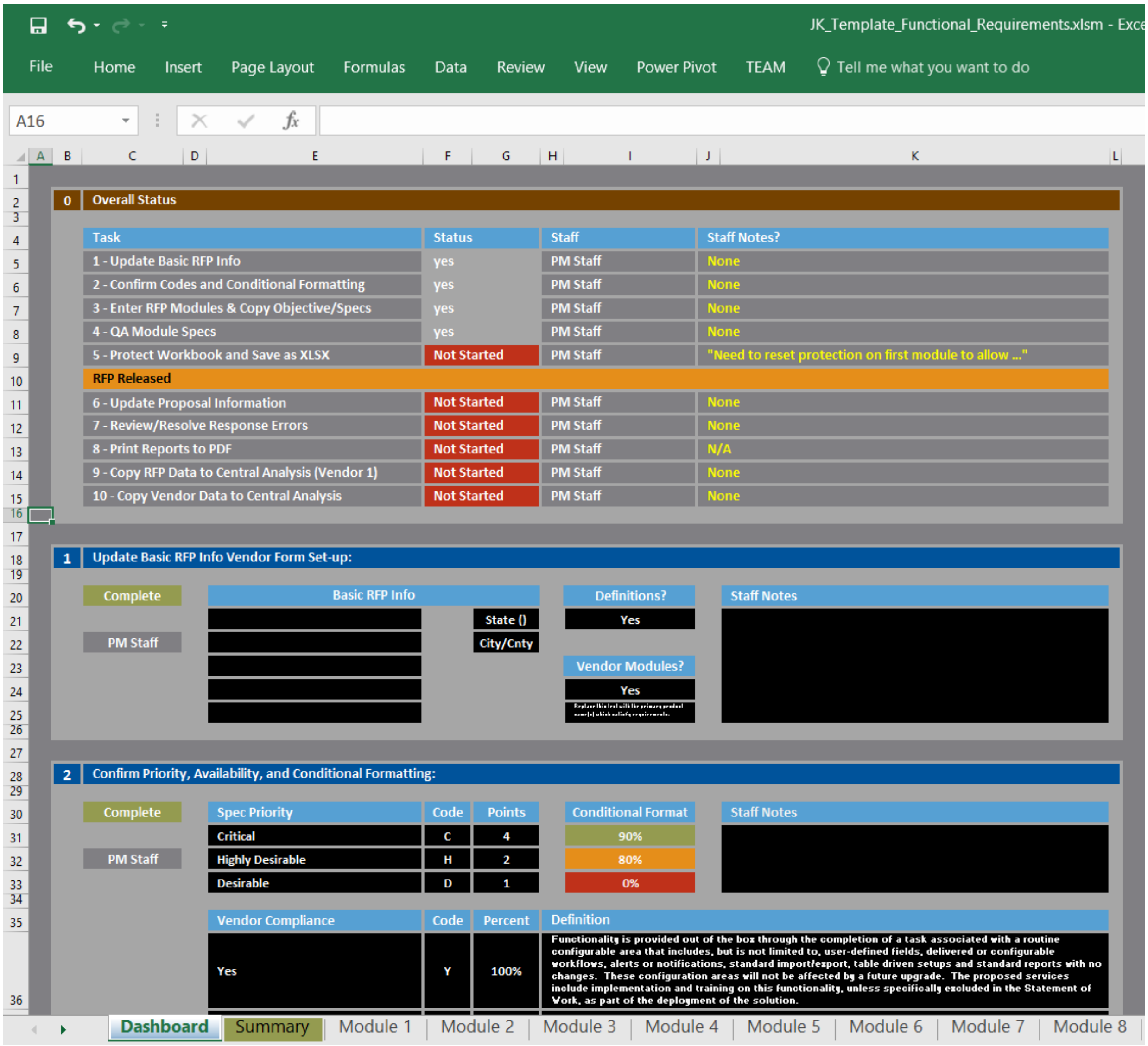
This "Functional Requirements Template" has three main components:

---

[1] Plante Moran is one of the largest certified public accounting and business advisory firm in the United States offering audit, accounting, tax and business advisory consulting services.

DASHBOARD SHEET

This sheet provides a summary overview of the completeness status of the information populated in the other sheets. It also enables a weighting to be applied to specific requirement areas that would then have a direct impact on results.



0	Overall Status		
Task	**Status**	**Staff**	**Staff Notes?**
1 - Update Basic RFP Info	yes	PM Staff	None
2 - Confirm Codes and Conditional Formatting	yes	PM Staff	None
3 - Enter RFP Modules & Copy Objective/Specs	yes	PM Staff	None
4 - QA Module Specs	yes	PM Staff	None
5 - Protect Workbook and Save as XLSX	Not Started	PM Staff	"Need to reset protection on first module to allow …"
RFP Released			
6 - Update Proposal Information	Not Started	PM Staff	None
7 - Review/Resolve Response Errors	Not Started	PM Staff	None
8 - Print Reports to PDF	Not Started	PM Staff	N/A
9 - Copy RFP Data to Central Analysis (Vendor 1)	Not Started	PM Staff	None
10 - Copy Vendor Data to Central Analysis	Not Started	PM Staff	None

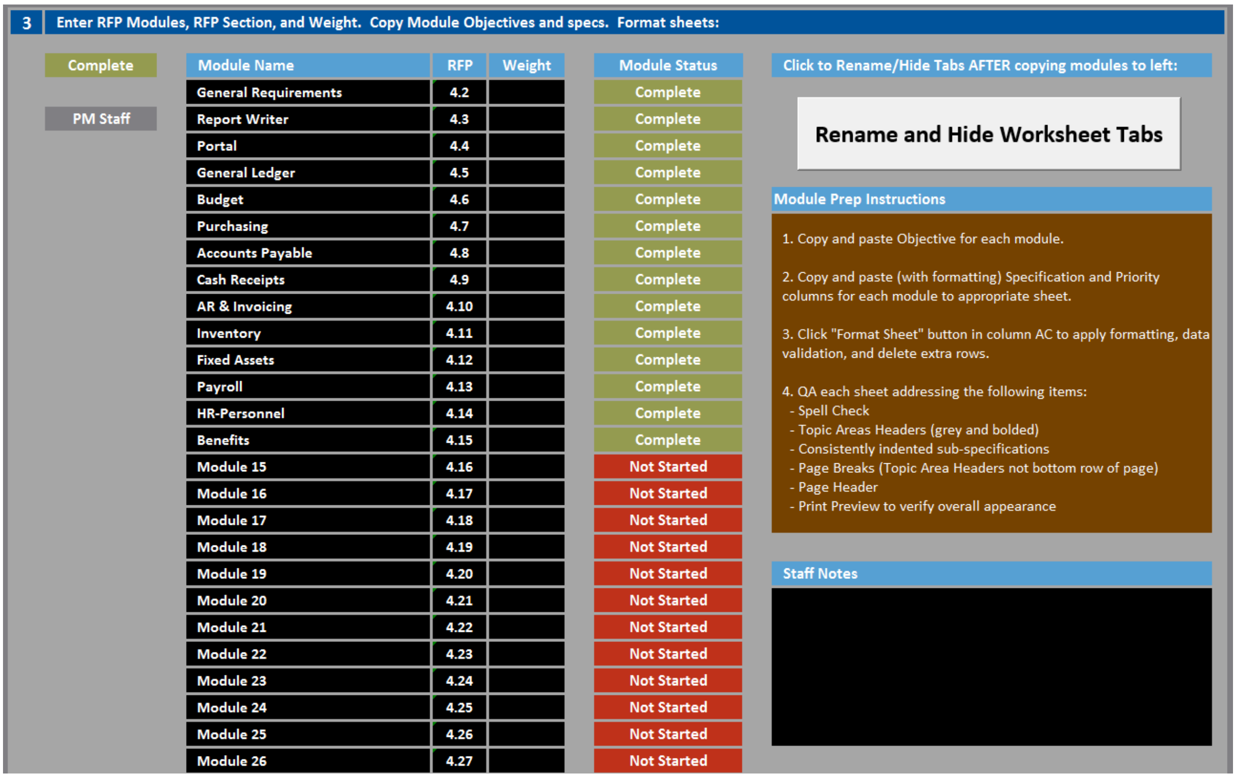
1 Update Basic RFP Info Vendor Form Set-up:

	Basic RFP Info		Definitions?	Staff Notes
Complete		State ()	Yes	
PM Staff		City/Cnty	Vendor Modules?	
			Yes	
			Replace this text with the primary product name(s) which satisfy requirements.	

2 Confirm Priority, Availability, and Conditional Formatting:

	Spec Priority	Code	Points	Conditional Format	Staff Notes
Complete	Critical	C	4	90%	
PM Staff	Highly Desirable	H	2	80%	
	Desirable	D	1	0%	

Vendor Compliance	Code	Percent	Definition
Yes	Y	100%	Functionality is provided out of the box through the completion of a task associated with a routine configurable area that includes, but is not limited to, user-defined fields, delivered or configurable workflows, alerts or notifications, standard import/export, table driven setups and standard reports with no changes. These configuration areas will not be affected by a future upgrade. The proposed services include implementation and training on this functionality, unless specifically excluded in the Statement of Work, as part of the deployment of the solution.
Reporting	R	75%	Functionality is provided through reports generated using proposed Reporting Tools.
Third Party	T	75%	Functionality is provided by proposed third party functionality (i.e., third party is defined as a separate software vendor from the primary software vendor). The pricing of all third party products that provide this functionality MUST be included in the cost proposal.
Modification	M	50%	Functionality is provided through customization to the application, including creation of a new workflow or development of a custom interface, that may have an impact on future upgradability.
Future	F	25%	Functionality is provided through a future release that is to be available within 1 year of the proposal response.
Not Available	N	0%	Functionality is not provided.



3 Enter RFP Modules, RFP Section, and Weight. Copy Module Objectives and specs. Format sheets:

Complete

PM Staff

Module Name	RFP	Weight	Module Status
General Requirements	4.2		Complete
Report Writer	4.3		Complete
Portal	4.4		Complete
General Ledger	4.5		Complete
Budget	4.6		Complete
Purchasing	4.7		Complete
Accounts Payable	4.8		Complete
Cash Receipts	4.9		Complete
AR & Invoicing	4.10		Complete
Inventory	4.11		Complete
Fixed Assets	4.12		Complete
Payroll	4.13		Complete
HR-Personnel	4.14		Complete
Benefits	4.15		Complete
Module 15	4.16		Not Started
Module 16	4.17		Not Started
Module 17	4.18		Not Started
Module 18	4.19		Not Started
Module 19	4.20		Not Started
Module 20	4.21		Not Started
Module 21	4.22		Not Started
Module 22	4.23		Not Started
Module 23	4.24		Not Started
Module 24	4.25		Not Started
Module 25	4.26		Not Started
Module 26	4.27		Not Started

Click to Rename/Hide Tabs AFTER copying modules to left:

Rename and Hide Worksheet Tabs

Module Prep Instructions

1. Copy and paste Objective for each module.

2. Copy and paste (with formatting) Specification and Priority columns for each module to appropriate sheet.

3. Click "Format Sheet" button in column AC to apply formatting, data validation, and delete extra rows.

4. QA each sheet addressing the following items:
- Spell Check
- Topic Areas Headers (grey and bolded)
- Consistently indented sub-specifications
- Page Breaks (Topic Area Headers not bottom row of page)
- Page Header
- Print Preview to verify overall appearance

Staff Notes

MODULE SHEET

The business requirements within the RFP are subdivided into various modules, commonly representative of business processes or departments. Within each module sheet, there is a list of requirements, against which the vendor (or the third-party advisor) can update with availability of function within solution, or indicate modification requirements and costs. The process is repeated for each requirement on the list on each module.

Replace this text with vendor name in the first module.

Code	Availability Definition
Y	Functionality is provided out of the box through the completion of a task associated with a routine configurable area that includes, but is not limited to, user-defined fields, delivered or configurable workflows, alerts or notifications, standard import/export, table driven setups and standard reports with no changes. These configuration areas will not be affected by a future upgrade. The proposed services include implementation and training on this functionality, unless specifically excluded in the Statement of Work, as part of the deployment of the solution.
R	Functionality is provided through reports generated using proposed Reporting Tools.
T	Functionality is provided by proposed third party functionality (i.e., third party is defined as a separate software vendor from the primary software vendor). The pricing of all third party products that provide this functionality MUST be included in the cost proposal.
M	Functionality is provided through customization to the application, including creation of a new workflow or development of a custom interface, that may have an impact on future upgradability.
F	Functionality is provided through a future release that is to be available within 1 year of the proposal response.
N	Functionality is not provided.

4.2 - General Requirements — Replace this text with the primary product name(s) which satisfy requirements.

Objective:

Number	Application Requirements	Priority	Availability	Cost	Required Product(s)	Comments	Deal Breaker?	Demo Item?
1	System must accommodate multiple investment entities, providing appropriate security for entity	C	Y	$100.00	N/A	Explain how this will be accomplished, functionality and technical setup		x
2	Access to, or updating of, information stored on the system should be restricted through use of a password security system at the User level.	C	Y	$50.00	N/A			x
3	Access to, or updating of, information stored on the system should be restricted through use of a password security system at the Workstation level.	C	R	$200.00	N/A			x
4	Access to, or updating of, information stored on the system should be restricted through use of a password security system at the Application level.	C	T	$100.00	N/A			x
5	Access to, or updating of, information stored on the system should be restricted through use of a password security system at the Transaction type, Account Code, Module, Department levels (add, change, inquire, delete) level.	C	T	$150.00	N/A			x
6	Ability for simultaneous users to access records	C	M	$30.00	N/A			
7	Ability to enter multiple transactions and requirements on a single screen.	C	F	$50.00	N/A			
8	Security should be managed at the company level.	C	F	$70.00	N/A			
9	Access to, or updating of, information stored in the database should be restricted through the use of an operating system or database management layer password security system for specific tables, rows, and columns by user for native or ODBC driver access to the database. System should allow for a multi-district environment.	C	N	$0.00	N/A			
10	Must provide a centralized database (not distributed), per District	C	Y		N/A			
11	Ability to limit access to "read only" at any level	C	Y		N/A			
12	Usernames should be unique to each individual; the capacity for authorized personnel to change passwords periodically should be provided. Should have self management on user passwords	C	Y		N/A			
13	All informational data elements tracked must be maintained in an integrated database to allow efficient data sharing, customized report writing, and automated posting.	C	Y		N/A			
14	The system must perform error checking to verify the quality of the information entered and that the system balances are maintained. Input validation	C	Y		N/A			x
15	The software should be web based and have a graphical user interface-based with comprehensive utility and "help" screen capabilities. Capability to drill down by user role, customized work list by role	C	Y		N/A			
16	Help screens should cover both functional reference and general user procedure.	C	Y		N/A		Y	x

Dashboard | Summary | Module 1 | Module 2 | Module 3 | Module 4 | Module 5 | Module 6 | Module 7 | Module 8 | Module 9 | Module 10 | Module 11 | Module 12 | Mod ...

SUMMARY SHEET

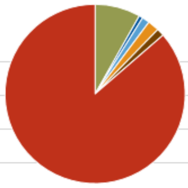
This sheet collates all the information from the various module sheets, once these have been updated, and presents a high-level summary showing solution compliance (mapping). Additionally, a graphical representation shows a breakdown of compliance, modification, interface etc. for each module.

Software Specification Compliance Summary				
		0		*Proposal Due Date: 01/00/1900*
0				*Date Received: Date Received*
		Vendor Long Name		
Section	**Module**	**Compliance**	**Weighting**	**Primary Product**
4.2	General Requirements	11.78%	0.00%	N/A
4.3	Report Writer	N/A	0.00%	N/A
4.4	Portal	N/A	0.00%	N/A
4.5	General Ledger	N/A	0.00%	N/A
4.6	Budget	N/A	0.00%	N/A
4.7	Purchasing	N/A	0.00%	N/A
4.8	Accounts Payable	N/A	0.00%	N/A
4.9	Cash Receipts	N/A	0.00%	N/A
4.10	AR & Invoicing	N/A	0.00%	N/A
4.11	Inventory	N/A	0.00%	N/A
4.12	Fixed Assets	N/A	0.00%	N/A
4.13	Payroll	N/A	0.00%	N/A
4.14	HR-Personnel	N/A	0.00%	N/A
4.15	Benefits	N/A	0.00%	N/A
	Weighted Average for Vendor Long Name:	**0.00%**	**Error <> 100%**	
	Weighted Average for Offered Modules:	**0.00%**		

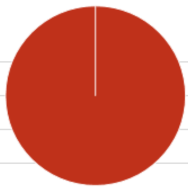
Total Potential Points Based on Specification Priority	
Points	**Description**
4	Critical Prioritized Requirements
2	Highly Desirable Prioritized Requirements
1	Desirable Prioritized Requirements

Percent of Total Potential Points Awarded Based on Vendor Availability Response		
Percent	**Response**	**Description**
100%	Y	Functionality is provided out of the box through the completion of a task associated with a routine configurable area that includes, but is not limited to, user-defined fields, delivered or configurable workflows, alerts or notifications, standard import/export, table driven setups and standard reports with no changes. These configuration areas will not be affected by a future upgrade. The proposed services include implementation and training on this functionality, unless specifically excluded in the Statement of Work, as part of the deployment of the solution.
75%	R	Functionality is provided through reports generated using proposed Reporting Tools.
75%	T	Functionality is provided by proposed third party functionality (i.e., third party is defined as a separate software vendor from the primary software vendor). The pricing of all third party products that provide this functionality MUST be included in the cost proposal.
50%	M	Functionality is provided through customization to the application, including creation of a new workflow or development of a custom interface, that may have an impact on future upgradability.
25%	F	Functionality is provided through a future release that is to be available within 1 year of the proposal response.
0%	N	Functionality is not provided.

4.2 - General Requirements				Overall Compliance:		11.78%
Availability	Priority			Total	Comments	Availability by Type
	Critical	ighly Desirabl	Desirable			
Yes	12			12	1	
Reporting	1			1		
Third Party	2			2		
Modification	3			3		
Future	2			2		
Not Available	111	12	4	127		
Total:	131	12	4	147	1	



4.3 - Report Writer				Overall Compliance:		N/A
Availability	Priority			Total	Comments	Availability by Type
	Critical	ighly Desirabl	Desirable			
Yes						
Reporting						
Third Party						
Modification						
Future						
Not Available	31		4	35	4	
Total:	31	0	4	35	4	



4.4 - Portal				Overall Compliance:		N/A
Availability	Priority			Total	Comments	Availability by Type
	Critical	ighly Desirabl	Desirable			

APPENDIX C – CASE DATA FOLDER STRUCTURE

This appendix outlines the business case data folder structures and related data manipulation files. These datasets are provided separately with this dissertation in digital format.

1. Raw_Data_Anon
    - CaseA_rawdata.xls
    - CaseB_rawdata.xls
    - CaseC_rawdata.xls
    - CaseD_rawdata.xls
    - CaseE_rawdata.xls
    - CaseF_rawdata.xls
    - CaseG_rawdata.xls
    - CaseH_rawdata.xls
    - CaseI_rawdata.xls
    - Raw_Data_Stats.xls

2. Raw_Data_Merged
    - CaseA_mergedata.xls
    - CaseB_mergedata.xls
    - CaseC_mergedata.xls
    - CaseD_mergedata.xls
    - CaseE_mergedata.xls
    - CaseF_mergedata.xls
    - CaseG_mergedata.xls
    - CaseH_mergedata.xls
    - CaseI_mergedata.xls

3. Raw_Data_Normalised
    - CaseA_normdata.xls
    - CaseB_normdata.xls
    - CaseC_normdata.xls
    - CaseD_normdata.xls
    - CaseE_normdata.xls
    - CaseF_normdata.xls
    - CaseG_normdata.xls
    - CaseH_normdata.xls
    - CaseI_normdata.xls

4. SCFUMv1
    - SCFUMv1.xlsx

5. SCF_Process_Node_Mapping
    - CaseA_nodemap.xls
    - CaseB_nodemap.xls
    - CaseC_nodemap.xls
    - CaseD_nodemap.xls
    - CaseE_nodemap.xls
    - CaseF_nodemap.xls
    - CaseG_nodemap.xls
    - CaseH_nodemap.xls
    - CaseI_nodemap.xls

6. SCF_Variance_Stats
    - Process_Node_Mapping_Stats.xls

7. SCFUMv2
    - SCFUMv2.xls

8. SCF_Process_Element_Mapping
    - CaseA_elemmap.xls
    - CaseB_elemmap.xls
    - CaseC_elemmap.xls
    - CaseD_elemmap.xls
    - CaseE_elemmap.xls
    - CaseF_elemmap.xls
    - CaseG_elemmap.xls
    - CaseH_elemmap.xls
    - CaseI_elemmap.xls

9. SCF_Variance_Mapping
    - NAV_VAR_SCFMap.xls

10. SCF_Full_Case_Mapping
    - CaseA_SCFUMv2map.xls
    - CaseB_SCFUMv2map.xls
    - CaseC_SCFUMv2map.xls
    - CaseD_SCFUMv2map.xls
    - CaseE_SCFUMv2map.xls
    - CaseF_SCFUMv2map.xls
    - CaseG_SCFUMv2map.xls
    - CaseH_SCFUMv2map.xls
    - CaseI_SCFUMv2map.xls
    - SCFUMv2_Mapping_Stats.xls

11. NAV_Mapping
    - NAV_Vanilla_SCFMap.xls

12. AX_Mapping
    - AX2012R3_TableList.xls
    - AX2012R3_SCFMap.xls
    - AX_Company_Accounts_Size.xls

13. ERP_SCI_Analysis
    - ERP_SCI_Analysis.xls

END

OF

DISSERTATION