Software Engineering: Knowledge and Performance

Narayanan SRINIVASARAGHAVAN©

A thesis
submitted in fulfilment of the requirements
for award of the degree of

Doctor of Philosophy
in
Information Sciences and Engineering

Faculty of Information Sciences and Engineering
University of Canberra
Canberra, Australia.

November 2010

Abstract

Software Engineering (SE) is a knowledge intensive activity. It is also a social process where individuals interact with each other in an organisational setting to achieve desired outcomes. The objective of this research is to explore the nature of the knowledge needed to conduct software development and the influence knowledge has on software development performance.

A survey of literature identifies that published works have attempted to examine the questions of knowledge and skill requirements to perform various roles in SE. However, this literature review reveals that while the need for some knowledge types is understood very well, other types have not been given adequate consideration. Further, it is important to understand knowledge and performance requirements based on solid foundational theories developed in fields such as cognitive psychology.

Cognitive theories on knowledge and action model how knowledge is stored, accessed and used in the human mind. On the other hand, the social origin theories of knowledge and expertise emphasise the importance of contextual or circumstantial knowledge to action. This research applies a reconciled model between these schools of thought to SE. Based on this model, both predominantly contextual and non-contextual cognitive knowledge are required to perform various roles in SE. A theoretical framework is developed that identifies nine types of knowledge required for an individual to perform SE roles. These knowledge types are 1) configuration knowledge (functional and technical knowledge about application systems), 2) social network knowledge, 3) process and procedural knowledge, 4) systems knowledge, 5) business domain area knowledge, 6) soft skills, 7) technique skills, 8) cultural and 9) heuristics knowledge. Using this framework, the research explores more broadly the knowledge required to perform various SE roles and examines the contribution that additional knowledge makes to perceived SE team performance improvement.

The theoretical framework has been tested using two research studies - first, using a case study of individuals in a software project setting and, secondly, using a survey of SE professionals to collect and analyse the knowledge requirements to perform their job. The analysis of survey data revealed a new classification schema of knowledge types for performing various SE roles. These are:

Factor 1: Knowledge of organisational culture and relationships Knowledge, Skills and Abilities (KSA) (Including organisational culture and organisational relationships)

Factor 2: Project software development KSA (Including technical skills, knowledge of application software items in the organisation and processes used in the SE team)

Factor 3: Management skills (Including negotiation, management and organisational skills)

Factor 4: Heuristic knowledge (including standards, guidelines, shortcuts and rules of thumb)

Factor 5: Communication and team work skills

Factor 6: Business functional KSA (Including business domain area knowledge, user interfaces, business analysis and testing)

Factor 7: Problem solving abilities (Working with others in the team, the ability to investigate, analyse, propose and solve problems and issues)

The results empirically demonstrate the nature of both contextual and non-contextual cognitive knowledge types that are required to perform various SE roles. The research identified the knowledge types that are significantly required in various phases of Software Development Life Cycle. It is found that technique skills and configuration knowledge contribute most to improvements in performance after an individual joins a SE team.

This research formulated, tested and found empirical support to theory of knowledge and performance that is based on both contextual and non-contextual knowledge types. By arriving at the knowledge classification and relationships this research has contributed to the theoretical knowledge base. The results of this study are significant to SE practice as they assist organisations in formulating knowledge management (KM) strategies. By implementing these KM strategies success rates can be improved, failures reduced and productivity enhanced. Further, the identified knowledge types are important to academia for better curriculum design on SE.

Table of Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank my supervisor Associate Professor Craig McDonald for mentoring, and guiding me through the long journey of research. He has been a source of encouragement, sound advice, constructive criticism and inspiration. Professor John Campbell started helping me in the second half of the research. I thank him for his encouragement, advice including on statistics and writing up the thesis.

My achievements, if any, in this research work is all founded upon my family. I owe it to my parents for building this passion for higher studies and research in me. Without the support, patience, sacrifices and constant encouragement from my wife, Sangeetha, this work would not have been possible. I am indebted to my brother for his help in seeing me through to higher studies. My sister has been a source of inspiration for me to complete the studies. I thank my parents-in-law and other family members for supporting me during the studies. My sincere love and thanks are owed to my children, especially my son, for their sacrifices and support.

The case study research was carried out in a major organisation. I thank the managers of this organisation for allowing me to conduct the case research. I thank the participants of the case study for contributing their invaluable time and insights for this research. I thank number of Data Collection Contacts (DCC) who helped me with collecting data for the general-survey. I would like to thank the numerous participants who filled the survey questionnaire. My work colleague Mr. Bruce Neindorf helped me improve my English by reviewing some of my earlier work during the research proposal stage. His help is very much appreciated.

Mr. David Pedderson, the UC statistical consultant, provided advice on statistical aspects of the research. I thank him for his valuable and insightful advice. I thank Ms Jennifer Bradley for the proof-reading and editorial support on the final thesis.

Narayanan SRINIVASARAGHAVAN

1 Introduction

1.1 Research objectives

This research is driven by a strong desire to build theory, by providing empirical evidence that is useful for practitioners and researchers in Software Engineering (SE) and knowledge management disciplines. The objectives of this research are:

- Identify the knowledge requirements to perform various SE roles
- Identify the knowledge factors that contribute to individual performance improvement in SE

The research seeks to apply theories of knowledge and action from psychology literature that is based on cognitive and social origins of knowledge into SE. In achieving these objectives, this research contributes to knowledge by adapting and applying existing theories into SE, by proposing a theoretical framework and testing it using a quantitative survey (referred to as general-survey in the thesis) and a case study in a real world SE team.

1.2 Background

1.2.1 Software Engineering (SE)

SE is the application of a systematic, disciplined, quantifiable approach to the development, operation and maintenance of software, in an engineering manner (IEEE, 2004). SE is an increasingly important activity in companies (Baskerville and Pries-Heje, 1999; Baskerville et al., 2007; Osterweil, 2007). Enterprises all over the world use various forms of software that have been tailored to their specific needs. There are different types of software such as business applications software and systems software (Pressman, 2009). Enterprises develop, maintain and support application software to support various business operations and functions within their organisations. Examples of business applications software include case and workflow management, accounting and core business applications. This research specifically investigates the research questions in the context of SE in enterprises. Software Engineering is a cognitive and team activity (Curtis et al., 1988; Rosson, 1996; Barthelmess and Anderson, 2002; Cubranic et al., 2004; He, 2004; Ye, 2006) where individuals work in an organisational social context to produce the outcomes (Gasson, 1998; Sawyer and Guinan, 1998; Gasson, 1999).

1.2.2 Knowledge

Knowledge is that which is known and the state of knowing (Machlup, 1980 and Plotkin, 1994, cited in Freeman, 2001). Davenport and Prusak (1998) say that individual knowledge is a fluid mix of framed experience, values, contextual information and expert insight that provides a framework for evaluating and incorporating new experiences and information. It originates and is applied in the minds of knowers (Davenport and Prusak, 1998). Rumizen (2002) says that knowledge is information in context that produces an actionable understanding.

Knowledge is broadly divided into two types: tacit and explicit. Tacit knowledge is personal, context-specific knowledge which is difficult to formalise, record or articulate and is usually in the heads of people (Tiwana, 2000). It is knowledge embedded in the human mind through experience and jobs (Awad and Ghasiri, 2003). On the other hand, explicit knowledge can be codified and transmitted in a systematic and formal language such as e-mail, documents or databases (Tiwana, 2000). Svieby (1997) likens tacit 1) knowledge to that of embodied knowledge and 2) explicit knowledge to that of theoretical knowledge. Software Engineering is a knowledge intensive activity (Glass, 1996; Robillard et al., 1998; Robillard, 1999; Dingsoyr, 2002; Ye, 2006; de Souza et al., 2007) requiring both tacit and explicit knowledge (Land et al., 2001; Philp and Garner, 2001; Forward and Lethbridge, 2002; Parsons and Saunders, 2004).

Operational knowledge refers to knowledge that is required to perform well in a given position in an organisation (Beazley et al., 2002). Such a kind of knowledge possessed by an expert maker (Dunne, 1993, cited in Chan, 2003) is referred to as Techne[1] by Aristotle (GMU, 2002) and has been adapted by Chan (2003) to model knowledge, skills and abilities in enterprise systems. Both tacit and explicit knowledge are required to perform well in a given position. Knowledge required to perform a job has been referred as Knowledge, Skills and Abilities (KSA) in the Human Resource Management (HRM) literature (Wooten, 1993; Summers and Summers, 1997; Schneider and Hunnius, 2003). This research investigates

1Aristotle calls techne a trained ability of rationally producing. i.e., the ability to produce something reliably under a variety of conditions, on the basis of some reasoning. This involves having knowledge, or having what seems to be knowledge (awareness) of whatever principles and patterns one relies on GMU (2002).

knowledge (in alignment to concepts of Techne and operational knowledge) that is required to perform a SE role and its contribution to improved performance.

1.3 Research questions

In line with the research objectives, the following research questions are examined:

1. What are the relevant types of knowledge required to perform various roles in an enterprise application SE context?
2. What are the contributing factors to improvement in individual performance from a knowledge point of view?

This thesis develops and tests a conceptual model of knowledge requirements to perform various roles in a SE context. The conceptual model is based on reconciliation between cognitive origin theories and social origin theories of knowledge and action. The reconciliation between these two theories highlighted the need for both contextual (specific to the organisation) and non-contextual knowledge to perform various roles in SE activities. This formulated model consists of nine types of job-specific knowledge types that contribute to performance. These are 1) configuration knowledge (functional and technical knowledge about the application system), 2) social network knowledge, 3) process and procedural knowledge, 4) systems knowledge, 5) business domain area knowledge, 6) soft skills, 7) technique skills, 8) cultural knowledge and 9) heuristics knowledge.

The first research question examines the need for the job-specific knowledge types identified above to perform various roles in SE. The relative difference in the need for these various knowledge types to perform various roles is examined as part of this question. The second research question examines the contribution of each of these knowledge types to improved performance of an individual working in a SE team.

1.4 Research motivation

SE is a knowledge intensive activity (Glass, 1997; Robillard, 1999; Dingsoyr, 2002; Desouza, 2003; Ye, 2006). Targeted knowledge management initiatives in SE (Glass, 1996; Canfora et al., 2002; Dingsoyr, 2002; Dingsoyr and Conradi, 2002; Hewitt and Walz, 2005; Petter et al., 2007; Aurum et al., 2008; Bjornson and Dingsoyr, 2008; Dingsoyr et al., 2009; Wolf et al., 2009) could improve success rates, improve productivity and reduce project failures. Hence, the research questions assume significance. Further, the research questions are of importance to theories in SE and Knowledge Management (KM) disciplines. Answers to these research questions will help professional practice to manage knowledge and improve success rates. This section discusses these research motivations in detail.

1.4.1 Improve success rates and reduce failures

Despite the use of best project management and software process methodologies, such as Capability Maturity Model Integration (CMMI) and Project Management Body of Knowledge (PMBOK), (Software Engineering Institute, 2004; Project Management Institute, 2009), as recommended by experts (Humphrey, 1995; Wasserman, 1996), SE projects fail (DeMarco, 1982; Philp and Garner, 2001; Ewusi-Mensah, 2003). Issues such as 1) schedule and budget overrun, 2) low quality and 3) functionality never delivered are typical (Ewusi-Mensah, 2003; Baskerville et al., 2007). An underlying theme of in-effective KM can be found in many of the failures. Software Engineering being knowledge work (Glass, 1997; Robillard, 1999; Dingsoyr, 2002; Desouza, 2003; Ye, 2006), an optimal management of knowledge is likely to result in success and reduce failures (Benson and Standing, 2001; Cooke-Davies, 2002; Ewusi-Mensah, 2003). Hence, it is important that we understand the knowledge requirements to perform various SE roles and its influence on performance.

1.4.2 Theory development

A sound theory explains a set of phenomena that specifies the key concepts that are operative in the phenomena and the laws that relate the concepts to each other (Jain and Boehm, 2005).

The need for sound theories in SE, based on empirical research has been highlighted in the literature (Glass, 1996; Rost, 2005; Jeffery, 2006; Osterweil, 2007; 2009). Many of the existing frameworks on knowledge requirements are primarily motivated by educational curricula design or HRM point of view (see section 2.2.5). These motivations lead to an incorrect understanding of knowledge requirements to perform a job, since the context of the job is not taken into account. Further, the identified knowledge requirements are not based on sound fundamental theory that has evolved over time in fields such as psychology (see section 2.2.5). While it is demonstrated in the literature that knowledge contributes to performance (Rasch and Tosi, 1992; Wade and Parent, 2001), further theoretical advancement need to be carried out on specific knowledge factors that contribute to improved performance. Thus, by answering the research questions on knowledge and performance this research contributes to SE theory.

1.4.3 Better curriculum design

Much research has been carried out to find the knowledge and skill requirements for performing SE roles, with the intention of developing better academic curricula at universities (Benbasat et al., 1980; Watson et al., 1990a; Lee et al., 1995; Bailey and Stefaniak, 2001). The existing gaps and the importance of aligning curricula with the needs of the profession have been stressed in literature (Lee et al., 1995; Tye et al., 1995; Bogoiavlenski et al., 1997). This research attempts to bring in a practitioner's perspective to understand the knowledge, skills and abilities required to perform various software roles. By answering the research questions, it is possible to develop better curricula, better training programs and better licensing requirements.

1.4.4 Better knowledge management in practice

KM is defined as the process by which an organisation generates wealth from its individuals and knowledge-based assets (Bukowitz and Williams, 1999). KM refers to an organization's capacity to gather information, generate knowledge, and act effectively and in an innovative manner on the basis of knowledge (Sage and Rouse, 1999). Numerous KM initiatives have been carried out in SE practice (Davenport, 1997; Henninger, 1997; Althoff et al., 2000; Klaus-Dieter et al., 2000; Dingsoyr, 2002; Mathiassen and Pourkomeylian, 2003).

Rus et al. (2001; 2002) find that KM needs to be carried out in SE organisations due to knowledge needs such as 1) acquiring knowledge about new technologies and accessing domain knowledge, 2) sharing knowledge about local policies and practices, 3) capturing knowledge and knowing *who knows what* and 4) collaborating and sharing knowledge. SE knowledge requires careful management (Devanbu and Brachman, 1991; Henninger, 1995; Henninger, 1997; Land et al., 2001; Ward and Aurum, 2004). In order to carry out such management of knowledge in software organisations, one needs to know the knowledge requirements to perform various roles in software organisations. It has been reported that even in SE organisations that have claimed to have implemented KM, the practices are still immature (Aurum et al., 2008). It was also found that management of tacit knowledge is given less importance in both SE research as well as practice (Bjornson and Dingsoyr, 2008). Only if we know the knowledge requirements, is it possible to carry out management of such knowledge. The first of the research questions attempts to answer the knowledge requirements to perform SE roles. Also, if we know the varying contributions of different knowledge types to increase in performance, we can focus our management activities to increase those types of knowledge. By answering the research questions it is possible to carry out targeted Knowledge Management initiatives in HRM activities and competency management. It is possible to manage knowledge in projects that are by nature temporary. These are explained in further detail.

1.4.4.1 Competency management

Competency management relates to identifying the KSAs required to perform various roles, in various positions, in organisations and have been prescribed in literature and in methodologies (Curtis et al., 1988; Nelson, 1991; Davenport, 1997; Davenport and Prusak, 1998). While these approaches prescribe competency management, they don't provide a framework of KSA required to perform various SE roles. This research is about identifying the KSA requirements to perform various roles in SE. The results of this research will be helpful for organisations that need to identify the competencies. By finding the relative contribution of knowledge factors to improved performance, it is possible to carry out focussed workforce competency management resulting in higher performance and productivity.

1.4.4.2 Recruitment, attrition and personnel rotation

Staff turnover may be linked to project failures (Hall et al., 2008). Difficulties in describing specific profiles for an ideal candidate (Nakayama and Sutcliffe, 2001) can exacerbate the problems due to staff turnover. The difficulties in describing specific profiles may be due to a lack of clear frameworks that aid IT managers on knowledge, skill and ability requirements to perform various roles. A well-defined framework, based on theory, will help in better management of staff turnover in scenarios such as recruitment, attrition and personnel rotation. Targeted training plans can be formulated to improve various types of knowledge of individuals in SE teams. This can improve the performance of the individuals as well as the team.

1.4.4.3 Project management and knowledge

It is known that project teams consist of individuals with a range of knowledge and skill sets (Klein et al., 1999). For instance, developer's knowledge influences productivity and progress of software development projects. (Yourdon, 1994, cited in Hanakawa et al., 2002). But a project is temporary in nature. This means a group of individuals are formed as a team to carry out specific work and are disbanded at the end. However, application software systems developed, enhanced or maintained by such teams live longer. When projects are disbanded, the knowledge walks out the door. It is not possible to explicitly capture the knowledge of experts who worked in the project team (Fischer and Ostwald, 2001). In the future, when maintenance, significant enhancement or change needs to be carried out to the application software system, the operational knowledge may not be available in the newly formulated SE team. The new team will have to undergo a learning process, thus sometimes re-inventing the wheel (Lytras and Pouloudi, 2003).

This research is about postulating and testing the relationship between knowledge and performance. The research attempts to identify the knowledge requirements to perform various roles within SE teams. The results of this research will be of enormous significance to project managers, who can act appropriately (such as estimate, plan, assess and track knowledge) to effectively manage the various types of knowledge of individuals within the team and organisations.

1.4.5 Summary

The research questions outlined in the thesis relating to 1) identifying knowledge requirements and 2) knowledge factors contributing to improved performance need to be answered for betterment of theory and practice. Development of sound theories between knowledge and performance is needed for the wider SE community. From a practice point of view, providing answers to the questions posed can help organisations better manage knowledge, thereby improving success rates, reducing failures and increasing productivity.

1.5 Chapter summary and Organisation of the thesis

The research seeks to answer the following questions:

1. What are the relevant types of knowledge required to perform various roles in an enterprise application SE context?
2. What are the contributing factors to the improvement in individual performance from a knowledge point of view?

There are a number of needs from practice, theory and academia that warrant the research questions to be answered in a scientific manner. By understanding the knowledge needs and their impacts on performance, sound theories can be built which contribute to knowledge. By applying such theories in real life SE environments using targeted knowledge management initiatives the performance can be improved. Theories on knowledge and performance in SE will allow educational institutions to develop better curricula and allow academicians to carry out the right training of students so that they are suitable for the industry.

This thesis has been structured into eight chapters (see organisation of thesis in Figure 1.1). The author of this research, as an SE practitioner, has a long term interest in investigating knowledge in SE. During the conceptualisation phase, the author undertook an extensive literature review and formulation of research questions and foundational theoretical framework for the research.

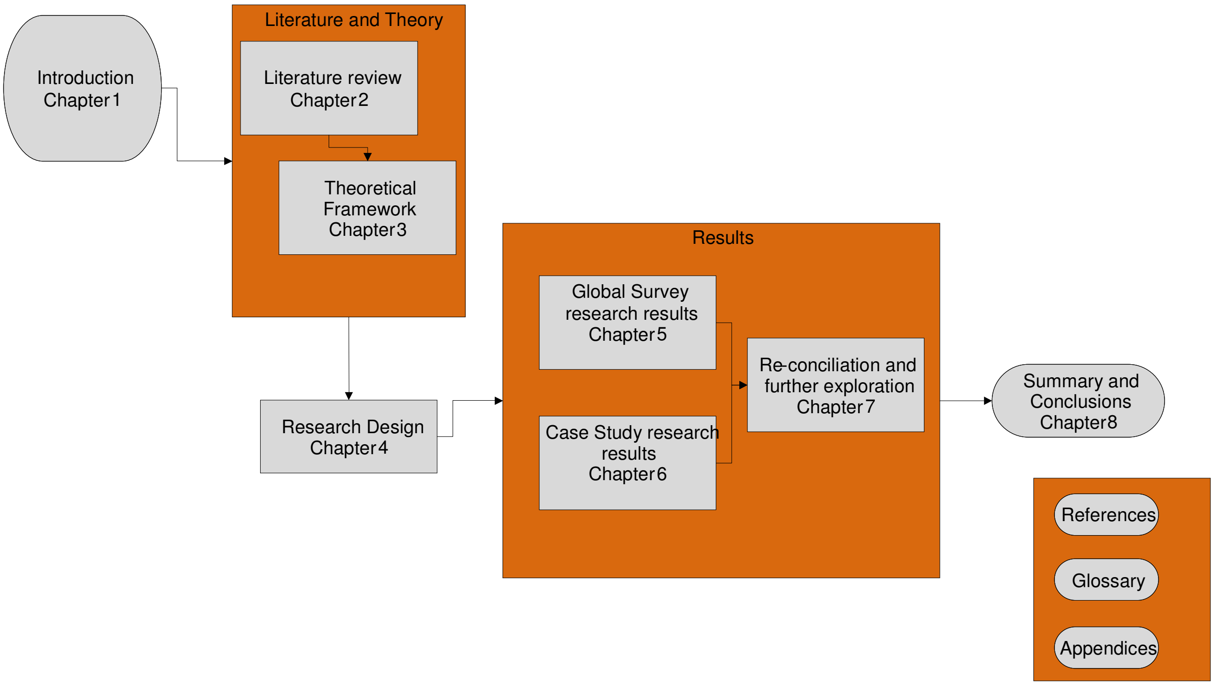

Figure 1.1. Flow chart: Organisation of the thesis

Chapter 1 discusses the questions that the research set out to answer. This chapter documents the need for answering these questions from both pure and applied research perspectives.

Chapter 2 reports the comprehensive literature review of work that has been carried out in the area of knowledge required for the SE discipline. It reports the work that has been carried out on knowledge contributions to performance. This chapter also identifies shortfalls in the existing literature.

Chapter 3 formulates a detailed theoretical framework for this research. This framework identifies various types of knowledge required to perform various roles in SE. This theoretical framework has been formulated by investigating theories in cognitive psychology and social origins of knowledge on knowledge and action.

Chapter 4 reports comprehensively the design for this research. A positivistic methodology that includes both a general-survey and a real-life case study is designed. This chapter describes the operationalisation of various factors identified in the original theoretical framework into operational items (operational variables). The chapter details the design of the case study and survey research methods such as sampling, data collection procedures and analysis plan.

Chapter 5 describes the general-survey conducted to find answers to the research questions. The chapter reports the analysis of data from the general-survey using various statistical methods. Using that data, this chapter answers both research questions on knowledge requirements to perform various roles as well as the knowledge factors that contribute to performance. The conclusions drawn are presented.

Chapter 6 reports the summary data and findings of the case study research carried out in a major organisation. The data gathered from the case study are analysed and conclusions drawn. The analysis contributes evidence to the proposed theoretical framework.

Chapter 7 integrates the findings from the case study research and the general-survey research and reconciles them. Numerous alignments exist between the findings of the research methods. The difference between the findings in some aspects is analysed and explained with possible reasons. The second part of this chapter explores a re-classification using Exploratory Factor Analysis (EFA) and reports the discovery of a transformed theoretical framework that is useful for both practice and theory.

Chapter 8 summarises the thesis and concludes by identifying the key research findings, contributions to knowledge for both practice and theory. The limitations of this research are discussed. The scope for future work is reported. Finally, the reflections of the researcher are presented.

2 Literature review

2.1 Introduction

Finding the knowledge and skill requirements for various Software Engineering (SE) roles has been a long standing aim of academia and industry. This aim has been motivated by desires such as developing the educational curriculum, training, human resource recruitment and macro-level skill planning. This literature review was carried out involving more than 700 research papers, 100 books and Ph.D theses. Studies relating to knowledge requirements for business application software and systems development fall between Information Systems (IS) and SE fields. (Pressman, 2009). Hence, the literature review included research papers on systems development from SE and IS areas.

This chapter reports the results of the literature review. Figure 2.1 shows the organisation of Chapter 2. Following this introduction, section 2.2 summarises the results of the literature review on the knowledge and skill required to perform various roles in SE. Section 2.3 presents the literature review on performance and how knowledge contributes to performance. Section 2.4 summarises the chapter.

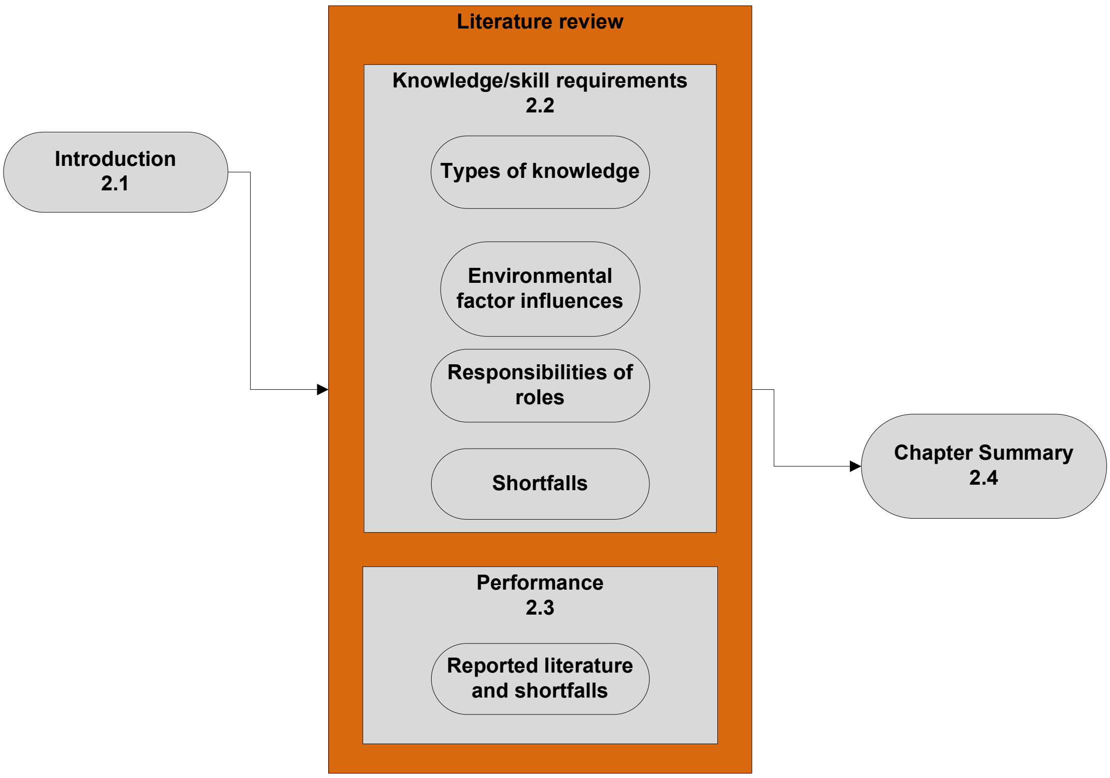

Figure 2.1. Flow chart: Organisation of Chapter 2

2.2 Knowledge/skill requirements

The literature review shows that several knowledge and skill requirements frameworks have been proposed over the years in the literature. A comparative summary of these frameworks is presented in Table 2.1. A summary of the literature is presented in Appendix section 11.15.

An iterative process involving literature review and formulation of theoretical framework was carried out which resulted in the identification of nine knowledge factors to perform SE roles. The process of deriving these categories is presented in Chapter 3. These are 1) Technical skills[2], 2) Soft skills, 3) Configuration knowledge (functional and technical knowledge of the application system), 4) Social network knowledge, 5) Business domain area knowledge, 6) Systems knowledge, 7) Process knowledge, 8) Cultural knowledge and 9) Heuristics knowledge. This literature review is organised around these nine categories.

2 Note that the term technique skills has been used in the theoretical framework developed in Chapter 3 instead of technical skills to cover a broader meaning for all roles (see section 3.5.2.2 for more details).

Purpose#2	Academic curriculum perspective		Knowledge Management Point of view			HRM Skills perspective				Based on recruitment – Job advertisements		Industry body of knowledge		
Type of knowledge	Henry et al. (1974 cited in Benbasat et al., 1980), McCubbray and Scudder (1988),	Benbasat et al.(1980), Monin and Dewe (1994)	Curtis et al. (1988), Herbsleb and Kuwana (1993),Walz et al. (1993) ,	(Chan, 2003), Cash (2004)	Zmud, (1983 pp.257-258), Nelson (1991),Nansi and Bennett (1998)	Lee et al. (1995), Sawyer et al (1998)	Leitheiser (1992)	Misic (1996)	Bailey and Stefaniak (2001)	Todd et al. (1995)	Wade and Parent (2001)	The Open group (2007b),	SWEBOK (IEEE, 2004)	Schwalbe (2002)
Technical skills	Computers	Generalist-Technical, Specialist-Technical	Technical skills	technical knowledge	general IS, technical skills	Technical skills, Technology management skills.	Analysis and design, programming, languages	technical skills	Technical skills	Technical knowledge	Technical skills	Enterprise Architecture Skills:, Technical IT Skills,	Knowledge areas such as design, construction, testing	Use of technology
Soft skills	People, Society,	Generalist-Managerial, Specialist-Managerial	Communication skills	Project management knowledge	organisational skills	Interpersonal management,	interpersonal,	communication skills and interpersonal skills.	soft skills	General management	organisational skills	Generic Skills,		Management skills, organisational skill
Social Network knowledge	Organisation		Social network knowledge	Relationship	organisational unit									
Cultural knowledge				company-specific knowledge	Organisational overview							Legal Environment		
Process knowledge													SE Process, SE quality	
Heuristics knowledge														
Business domain area knowledge	Organisation		Application domain knowledge	business knowledge		Business functional skills	business, environment		business concept skills	Business knowledge		Business Skills, Legal Environment		
Application system knowledge			Application system knowledge, scenarios of use	product-specific knowledge,	IS product									
Systems knowledge	Systems, Models				general IS			Analytical skills		Systems knowledge				

#1 – The knowledge identified by these authors have been combined together as they have similar concepts discussed.

#2 – Grouping based on primary purpose.

Table 2.1. Key classifications on knowledge and skill requirements #1

2.2.2 Knowledge Types

2.2.2.1 Technical skills

Technical skills are defined as those skills acquired through training and education or learned on the job and are specific to each work setting (Litecky et al., 2004). Technical skills encompass knowledge of computing infrastructures including hardware, distributed technologies, programming languages, systems integration, computer networks, internet/intranet development and support, desktop support, data communication and databases (McCubbray, 1988; Nelson, 1991; Lee et al., 1995; Bailey and Stefaniak, 2001). Examples of technical skills include 1) Web programming, 2) Unix, 3) C++, 4) Java, 5) Oracle database 6) SQL programming, 7) C++, and 8) Visual Basic (McCubbray, 1988; Nelson, 1991; Lee et al., 1995; Bailey and Stefaniak, 2001).

Due to the technological nature of SE, the need and emphasis has been on technical skills for a long time (Benbasat et al., 1980; Cash et al., 2004). A review of the Table 2.1 shows that all the identified knowledge and skill frameworks have recognised the need for technical skills. Content analysis of job advertisements demonstrates that technical skills appear most in recruitment advertisements (Lee et al., 1995; Todd et al., 1995; Nansi, 1998; Gallivan et al., 2004; Litecky et al., 2004; Prabhakar et al., 2005). Technical skill requirements vary over periods of time (Leitheiser, 1992; Monin and Dewe, 1994) and for specific organisations (Sawyer et al., 1998). Several academic works have demonstrated that SE roles including project managers, architects, business analysts, designers, programmers and testers require technical skills to perform their jobs (Lee et al., 1995; Todd et al., 1995; Nansi, 1998; Gallivan et al., 2004; Litecky et al., 2004; Prabhakar et al., 2005). Programmers and systems analysts have been found to need technical skills at high levels when compared to other non-technical skills (Cheney and Lyons, 1980; Cheney, 1988; Misic, 1996), while project managers require them at a lower level (Nansi and Bennett, 1998). In summary, substantial amount of work has been carried out and reported in literature that demonstrates the need for technical skills to perform SE roles.

2.2.2.2 Social network knowledge

Social network knowledge is an understanding of the crucial relationships including the structure, interpersonal dynamics within the structure, common context and language held by individuals, that are required to perform the job (Lesser, 2000; Beazley et al., 2002; Beazley et al., 2003). KM literature has understood the importance of social network knowledge to performance (Lesser, 2000; Anklam, 2003; Cross and Parker, 2004; Krebs, 2008). It should be clarified here that there is a difference between the ability to social network and the actual social network knowledge. Ability to social network is a soft skill. On the other hand, social network knowledge is about the existing relationships. Ability to social network is a skill one possesses with which they gain the actual social network knowledge.

There has been limited recognition of social network knowledge in knowledge and skill frameworks that have been proposed early on. Benbasat et al., (1980) found that 'Knowledge of society' to be important for managers and systems analysts. Zmud (1983) recognised the need for social network knowledge to a limited extent through a category on *Organisational unit*, which included an understanding of links with other internal and external units. Such a need to understand the organisational structures is also recognised by Cash et al., (2004).

Walz et al., (1993) found that the knowledge is shared amongst members of the SE team showing the importance of social network knowledge. Based on studies of six systems managers, Ives and Olson (1981) showed that 76% of the time of managers is spent in oral communication. Curtis et al., (1988) found that large number of groups had to coordinate their activities, or at least share information during software design. They found that a software engineer would normally communicate most frequently with team members, slightly less frequently with other teams on the project, much less often with corporate groups and except for rare cases, very infrequently with external groups (Curtis et al., 1988).

Content analysis of newspaper advertisements (such as Todd et al., (1995) Shi Nansi (1998). Wade and Parent (2001)) do not show up social network knowledge requirements to perform SE roles since job advertisements represent only an initial screening process (Hunter and Palvia, 1996). Schwalbe's (2002) list of skills for IT project managers does not cover the social network knowledge. Frameworks, such as SE Body Of Knowledge (SWEBOK) (IEEE, 2004) (see Appendix section 11.13 for details), and TOGAF (The Open group, 2007a) on knowledge and skill requirements for various roles have not considered the need for social network knowledge to perform various roles.

While some recognition for social network knowledge is provided in some of the literature (Benbasat et al., 1980; Zmud, 1983; Curtis et al., 1988; Walz et al., 1993), other significant quantitative studies and frameworks on knowledge and skill requirements (Leitheiser, 1992;

Lee et al., 1995; Sawyer et al., 1998; IEEE, 2004; The Open group, 2007b) have not considered social network knowledge as a factor (see Table 2.1). Thus, it is possible to say that evidence and discussion regarding the need for social network knowledge to perform various SE roles is at the least patchy. The literature has given weak emphasis to the importance of social network knowledge in performing SE roles. The Body of Knowledge that are published by industry bodies have not considered the need for social network knowledge. These problems are discussed in section 2.2.5.

2.2.2.3 Soft skills

Ehrlich (2004) defines soft skills as the skills needed to perform jobs where job requirements are defined in terms of expected outcomes, but the processes used to achieve the outcomes may vary widely. Soft skills refer to non-technical skills performed in the intra- and inter-personal domains that facilitate the application of technical skills and knowledge (Kantrowitz, 2005). Soft skills are generic skills, typically comprising leadership, team working and inter-personal skills (The Open group, 2007b). The comparative review shown in Table 2.1 shows that most of the identified knowledge and skill frameworks have recognised the need for soft skills.

Henry et al., (1974, cited in Benbasat et al., 1980) showed that there was a general need for more soft skills to perform various SE roles. Benbasat et al., (1980), McCubbray and Scudder (1988) found that generalist skills were important in performing the job of managers as well as systems analysts. Curtis et al., (1988) found that communication and coordination breakdowns were a problem in the design of large software systems, showing the importance of soft skills. Zmud's (1983) classification[3] includes organisational skills (such as interpersonal behaviour, group dynamics and project management) as an important part of his framework. Lee et al., (1995) and Sawyer et al., (1998) found the need for interpersonal management skills as important to performing system development roles. Monin and Dewe (1994) based on a study at New Zealand found that people skills, verbal and presentation skills and written communication skills were the top three skills of significance to various systems jobs. Watson et al. (1990b) and Misc (1996) found that communication and interpersonal skills were more important for system analysts than programmers. Becker et al., (1997) demonstrated that communication skills were perceived to be significantly more

[3] empirically validated by Nelson (1991) and Nansi and Bennet (1998)

important for managerial employees than for technical employees. Orsted (2000) states that soft skills are focussed during recruitment for all types of roles at Microsoft.

Soft skill types have been demonstrated to be important in performing the job of a programmer (Bailey and Stefaniak, 2001), business analyst (Evans, 2004), project manager (Ives and Olson, 1981; 2002), architects (Framption et al. 2005 cited in Frampton et al., 2006; The Open group, 2007b) and systems analysts (Vitalari and Dickson, 1983; Hunter, 1994; Hunter and Palvia, 1996). Content analysis of job advertisements demonstrates that soft skills appear in recruitment advertisements for different roles (Lee et al., 1995; Todd et al., 1995; Nansi, 1998; Wade and Parent, 2001; Gallivan et al., 2004; Prabhakar et al., 2005).

Thus, the literature survey indicates that there is both quantitative and qualitative evidence of the need for soft skills such as communication skills, organisational skills to performing various SE roles.

2.2.2.4 Application system knowledge[4]

SE is about building, enhancing or maintaining an application software system in an enterprise context. Application system knowledge refers to the functional and technical knowledge of the specific software application(s), which is the focus of the SE activity[5]. Application system knowledge is largely contextual to the organisation. Hence, this knowledge gains significance due to issues such as individuals changing jobs frequently (Rilling, 2002). Experts with wealth of application system knowledge perform well in SE roles and occupy high-profile positions within their SE teams (Curtis et al., 1988; Sawyer et al., 1998). Several field studies have demonstrated the need for application system knowledge to perform various SE roles (Curtis et al., 1988; Curtis et al., 1990; Herbsleb and Kuwana, 1993; Walz et al., 1993).

Walz et al. (1993), based on a field study, found the need for application system knowledge across the software development staff. They recommended assigning individuals possessing application system knowledge to a design project can reduce the learning time involved (Walz

4 Also referred to as configuration knowledge in this thesis (see 3.5.2.3 on why the term was chosen)

5 Business domain knowledge and application system knowledge need to be differentiated since some studies, notably Curtis (1988; 1990), mixed them up. Business domain knowledge is an understanding of business domain such as insurance or banking and is different from application system knowledge (Herbsleb and Kuwana, 1993).

et al., 1993). Zmud (1983) outlined the need for application system knowledge by categorising system product knowledge (including purpose, design, required procedures, impacts on individuals) in his six knowledge type framework. However, subsequent testing of Zmud's framework by Nelson's (1991) and Nansi and Bennet (1998) had validity issues[6].

While many qualitative field studies (Curtis et al., 1988; Curtis et al., 1990; Herbsleb and Kuwana, 1993; Walz et al., 1993; Rilling, 2002) provide evidence of the need for application system knowledge, it is surprising to note that numerous other quantitative studies (Cheney and Lyons, 1980; Cheney, 1988; Leitheiser, 1992; Monin and Dewe, 1994; Lee et al., 1995; Sawyer et al., 1998) that were carried out to find knowledge and skill requirements of SE roles did not consider application system knowledge as a factor (see Table 2.1). Further, the Body of Knowledge published by industry bodies have not considered the need for application system knowledge (see section 2.2.5.1.).

Content analysis of newspaper advertisements (such as Todd et al., (1995) Shi Nansi (1998) and Wade and Parent (2001)) suffer from a disadvantage in detecting the need for application systems knowledge. Application systems knowledge is organisation specific. During recruitment, it is not possible to look for this knowledge from outsiders. Hence, it can be expected that the need for application systems knowledge won't be published in the newspaper advertisements.

In summary, it is seen that there is qualitative evidence (Curtis et al., 1988; Curtis et al., 1990; Herbsleb and Kuwana, 1993; Walz et al., 1993) in the literature to suggest that application system knowledge is important to performing various SE roles. Some of the quantitative studies that tested application system knowledge suffer from some short comings and cannot be relied upon fully. Many other studies (Cheney and Lyons, 1980; Cheney, 1988; Leitheiser, 1992; Monin and Dewe, 1994; Lee et al., 1995; Sawyer et al., 1998) have not considered

---

6 Nelson (1991) grouped both users and IS professionals into one unit in his survey. Users and SE professionals view the knowledge needs differently (Green, 1989). Further, Nelson's work (1991) operationalised the application system knowledge from the point of view of users of application systems. Hence, Nelson's (1991) operational items for application system knowledge suffers from construct validity issues. Nansi and Bennet (1998) changed the definition provided by Zmud's (1983) for IS product knowledge. Nansi and Bennet (1998) introduced the word market place in the definition leading to a totally different meaning. Hence, the result of these studies (Nelson, 1991; Nansi and Bennett, 1998), with respect application system knowledge, is questionable.

application system knowledge at all. Hence, at the best, it is possible to say that weak emphasis has been provided to application system knowledge in the literature.

2.2.2.5 Business domain area knowledge

SE is about building, enhancing or maintaining an application software system in an enterprise context such as insurance, banking or airline. The general knowledge of the business domain area of the application being developed is referred to as the business domain area knowledge (Bailey and Stefaniak, 2001) and includes functional knowledge about business aspects of the organisation (Chan, 2003), grasp of business issues, and an understanding of the environment in which the business exists (Lee et al., 1995; Cash et al., 2004). The comparative review shown in Table 2.1 shows that many of the identified knowledge and skill frameworks have recognised the need for business domain area knowledge skills.

Several reported works in the literature have recognised the need for business domain area knowledge to perform various roles in SE. Curtis et al.[7], (1988) found that deep application domain knowledge required to build most complex software systems was thinly spread through development staff. Through experimentally created design contexts, Adelson and Soloway (1985) demonstrated that the contextual business domain area knowledge is central to designers. The importance the need for modelling of domain knowledge to SE has been emphasised by Iscoe et al.(1991) and Fischer (1996). Herbsleb and Kuwana (1993) suggested that application domain knowledge be captured and preserved during design for later stages of development.

Walz et al., (1993) found that the length of time required for requirements analysis phase is dependent on the breadth and depth of application domain knowledge the team members bring to the project. Lee et al. (1995) found the importance of business functional skills for performing various systems development roles. Further empirical support to the need for business functional knowledge and skills, at a large organisation, was provided by Sawyer et al., (1998) and Cash et al. (2004). Bailey and Stefaniak (2001) found that business domain area knowledge is required to perform the job of a programmer. Frampton et al., (2005, cited

7 Curtis et al., (1988) grouped both application system knowledge as well as business domain are knowledge into a single construct titled Application domain knowledge.

in Frampton et al., 2006) say that business orientation that includes an understanding of business domain knowledge is one of the capabilities identified for practicing IT architects. A business analyst role is about analysing needs, processes and requirements of the business. Hence, it can be expected that they need to possess business domain knowledge (Evans, 2004). Schwalbe (2002) suggests that IT project managers need to have knowledge of the industry. Todd et al., (1995) found, using content analysis of newspaper advertisements, that functional knowledge of business environment is important for performing SE roles. Roy Chan's (2003) knowledge modelling for enterprise systems showed that business domain knowledge is important to performing various roles, such as project manager and programmer.

Thus, the literature survey shows that there is adequate evidence to suggest the importance of business domain knowledge to performing various SE roles.

2.2.2.6 Systems knowledge

Systems knowledge is defined as an understanding of the interplay of cause-and-effect relationships that is essential for sound decision making (Beazley et al., 2002). It includes problem-solving skills and analytical and modelling skills as well as knowledge of development methodologies (Todd et al., 1995).

The need for systems knowledge has been understood in early studies on knowledge and skill requirements. Benbasat et al. (1980) found that systems knowledge is important for systems analysts and managers. Cash et al., (2004) found that the ability to interpret business problems and develop solutions is a significant competency that is required more in the e-commerce age. However, the Zmud's (1983) categories of knowledge required to perform information system implementation did not include any category for systems knowledge such as problem solving or analytical abilities. This notable omission has continued in the work by Lee et al., (1995) and Chan (2003) as well. However, Sawyer et al., (1998), who followed on work by Lee et al., (1995), based on the analysis of qualitative data by codifying the themes from interview and focus groups, found that systems knowledge including combination of skills to be a problem solver was prevalent and required to perform system development roles.

Orsted (2000) suggests that problem solving skills are one of the key skills expected of a new recruit at Microsoft. Vitalari and Dickson (1983) say that highly rated systems analysts exhibit behaviours that enable them to solve problems such as analytical reasoning, planning goal setting and strategy formulation skills. Misic (1996) found that analytical skills are important for systems analysts. Nord and Nord (1997) found that the need for system skills is not different between experienced and less-experienced system analysts. Bailey and Stefaniak (2001) demonstrated problem-solving skills as important to perform the job of a programmer.

Frampton et al., (2006) based on a survey found that the certified architects perceived themselves as being better problem solvers than uncertified architects. Based on a content analysis study of newspaper advertisements, Todd et al.,(1995) found that references to systems knowledge increased in newspaper advertisements, since 1970, for programmers. However, for systems analysts and project managers references to systems knowledge were mentioned less frequently than other categories such as 1) technical skills and 2) business knowledge (Todd et al., 1995) and could be due to limitations[8] of content analysis. The comparative review shown in Table 2.1 shows that many of the identified knowledge and skill frameworks have recognised the need for systems knowledge. In summary, it is found that there is adequate evidence to show the need for systems knowledge to performing various SE roles.

2.2.2.7 Process knowledge

The importance of well defined processes to SE efforts has been well understood in professional practice and by academia. The importance of following processes in SE has been emphasised in literature (Humphrey, 1995; Wasserman, 1996). Many organisations have implemented process maturity frameworks such as CMMI (Software Engineering Institute, 2004) and SPICE (ISO-SPICE, 2008). The guide to SWEBOK includes the SE process knowledge area. This knowledge area includes the definition, implementation, assessment, measurement, management, change and improvement of the SE process (IEEE, 2004).

Despite widespread acceptance of the need for SE processes, a number of studies that attempted to find the knowledge and skill requirements did not consider process knowledge requirements to perform various roles (Cheney and Lyons, 1980; Cheney, 1988; Leitheiser, 1992; Monin and Dewe, 1994; Lee et al., 1995; Sawyer et al., 1998).

---

[8] Newspaper advertisements only represent an initial screening process and hence do not contain the full skill requirements of the position. (Hunter and Palvia, 1996). Further, dummy ads may be placed for legal reasons (Gallivan et al., 2004).

Perhaps due the predominantly organisational specific nature of process knowledge, the content analysis studies (Lee et al., 1995; Todd et al., 1995; Wade and Parent, 2001) have not identified the need for process knowledge to perform the jobs. The TOGAF (The Open group, 2007a) skill framework discusses the need for business process knowledge for architects. This view is reinforced in the study by Chan (2003) on enterprise systems. However, neither TOGAF nor Chan explicitly identified the need for SE process knowledge for performing various roles. The comparative review shown in Table 2.1 shows that many of the identified knowledge and skill frameworks have not recognised the need for process knowledge.

Despite many of the literature not identifying process knowledge as needed to perform SE roles, it is a knowledge that needs to be considered importantly. Large SE organisations generally have many documented and defined processes. If processes are important for SE effort (Humphrey, 1995; Wasserman, 1996) then an understanding of these processes amongst the SE professionals is also required to properly implement them. While the literature clearly outlines the need for following processes in SE, the importance of process knowledge to performing SE roles has been given a weak emphasis in the literature.

2.2.2.8 Cultural knowledge

Larry Samovar and Richard Porter (1995) have defined culture as the deposit of knowledge, experience, beliefs, values, attitudes, meanings, hierarchies, religion, notions of time, roles, spatial relations, concepts of the universe, and material objects and possessions acquired by a group of people in the course of generations through individual and group striving. Culture is a knowledge asset for the organisation and is embedded in social processes, institutional practices, traditions and technologies, artefacts (Boisot, 1998). The KM literature (Davenport, 1997; Awad and Ghasiri, 2003) recognises the importance of culture to managing knowledge.

Telliglu and Wagner (1999) found that SE practice depends on context and culture. The issues relating to communication and coordination breakdowns in large design projects as identified by Curtis et al., (1988) could be related to lack of cultural knowledge amongst the individuals. Cash et al., (2004) identified that in the e-commerce era many competencies that are based on an understanding of culture were required, such as 1) more fluid interactions, 2) greater levels of uninhibited interaction, 3) knowledge of strategy risks, 4) opportunities and 5) the ability to understand the business environment. It was found that an understanding of business culture is required to perform the role of programmers (Bailey and Stefaniak, 2001). Chan (2003) found evidence for cultural, social norms and practice knowledge requirement to performing various roles in enterprise system development.

Organisational cultural knowledge is largely tacit (Mandl et al., 2008) and contextual and hence is difficult to articulate in newspaper advertisements during recruitment. Hence, none of the content analysis of newspaper advertisements explicitly recognised the need for culture in performing various SE roles.

While there are some indications of the need for cultural knowledge to perform SE roles (Curtis et al., 1988; Bailey and Stefaniak, 2001; Cash et al., 2004), many other studies did not consider its need at all (Cheney and Lyons, 1980; Cheney, 1988; Leitheiser, 1992; Monin and Dewe, 1994; Lee et al., 1995; Sawyer et al., 1998). While Zmud's (1983) classification scheme for knowledge needs included some connections to cultural knowledge items[9], he failed to include an explicit knowledge factor for cultural knowledge. The guide to SWEBOK does not include cultural knowledge. In summary, while the literature has touched upon the need for cultural knowledge to perform SE roles, limited empirical evidence is available in the literature.

2.2.2.9 Heuristics knowledge

Heuristics means a useful shortcut, an approximation or a rule of thumb for guiding search, such as a strategy that a chess master uses to reduce the enormous space of possible moves at each point in a game (Gigerenzer and Todd, 1999). Vitalari and Dickson (1983) provided evidence that some types of heuristic problem solving behaviours lead to better performance in systems analysts. Zmud's (1983) categorisation of knowledge items related to the need to know the policies of the organisation some of which can be considered to be heuristics in nature (Busch et al., 2001). Chan (2003) found that company specific policies and rules are important to performing various roles in Enterprise System development.

While some literature notes the need for some types of heuristics knowledge, such as shortcuts and policies (Vitalari and Dickson, 1983; Zmud, 1983; Chan, 2003), more broadly

9 Organisational unit knowledge (including objectives, purpose, functions, resources, links and problems to perform IS roles) and organisational over view knowledge (objectives, purpose, opportunities, constraints, internal and external functioning).

the literature surveyed did not give adequate consideration to the need for heuristic knowledge (Cheney and Lyons, 1980; Cheney, 1988; Leitheiser, 1992; Monin and Dewe, 1994; Lee et al., 1995; Sawyer et al., 1998) to perform SE roles. The comparative review shown in Table 2.1 shows that none of the identified knowledge and skill frameworks have explicitly recognised the need for heuristics knowledge. Perhaps due to its more tacit nature, frameworks, such as SWEBOK, TOGAF (The Open group, 2007a), have not given explicit significance for heuristics knowledge in their skill frameworks. Thus, while there is some recognition that heuristics knowledge plays a part in performing roles in SE, the literature has not tested and provided empirical evidence of the need for heuristic knowledge.

2.2.3 Relative importance of knowledge types

Numerous studies have attempted to compare the relative need for different knowledge types with reference to SE and IS roles. Benbasat et al., (1980) found that managers and systems analysts perceived a greater usefulness for generalist skills than for specialist skills. Nelson (1991) identified that personnel were most deficient in the area of general systems knowledge and in need of more organisational knowledge. Nansi and Bennet (1998) based on a study of systems managers in Singapore found that technical skills were the lowest ranked skills required to perform their job. Monin and Dewe (1994) found that people skills, verbal presentation skills and written communication skills were the top three skills of significance to various systems jobs in New Zealand. This was followed by keeping abreast with new technology, and knowledge of the business area. They found a shift from technology related skills requirement to business-related skill requirements in New Zealand. Misic (1996), based on a survey, found that communication and interpersonal skills were the top ranking skills needed followed by technical skills for system analysts. Analytical skills were ranked with the lowest need. Wong et al. (2000, cited in Gallivan et al., 2004) suggested that non-technical skills continued to be more important than technical skills.

There are number of problems in making comparisons between different skill types. The studies reported in the literature that compare technical and non-technical skills have not approached the knowledge and skill requirements from the point of view of performing the role (See section 11.15) instead from a Human Resource Management (HRM) viewpoint. Some knowledge types (configuration knowledge, process knowledge, cultural and social network knowledge and heuristics knowledge) have not been adequately considered in the existing literature. These gaps in the literature mean that any conclusion on relative importance of technical and non-technical skills is still questionable.

2.2.4 Other considerations

2.2.4.1 Environmental factors

The knowledge requirement for performing various roles is dependent on environment factors as well. Research studies have highlighted the differences in skill requirements, job performance and other factors between men and women. Igbaria and Chidambaram (1997), based on a survey, found significant gender effects on work and life experience, organisational level, salary, boundary spanning activities and intention to stay. Gallivan (2004), based on a multi-method study, found partial support for the hypothesis that gender will be related to IT professionals' non-technical skills with women perceived as having stronger non-technical skills than men. There are differences between men and women in terms of listening and communication skills (Brownwell, 2002). For example, women are thought to be more sensitive to non-verbal cues than men, with men performing better in terms of visual skills (Brownwell, 2002). Based on a survey, it has been reported that perceptions and preferences of skills are related to gender and age groups (Lerouge et al., 2005).

Benbasat et al. (1980) found that organisational maturity has impacts on the skill requirements to perform various roles. Green (1989) based on a survey with 872 responses from both public and private sector system analysts found that the perceived skill requirements for systems analysts are different between public and private sector organisations. It was found by Sawyer et al. (1998) that the corporate IT skill levels, both current and the perceived skills needed for future, are affected by both respondent age and level of IT experience. Nord and Nord (1997) found that there was no statistically significant relation between experience and perceived knowledge and skill requirements for system analysts. The above evidence suggests to us that the perceived skill requirements are dependent on the contextual and environmental factors within organisations.

2.2.4.2 Responsibilities of roles

The roles of system analyst, programmer-analyst, project manager, business analyst, testers, architects and designers are commonly found in the literature. Since, the knowledge and skill requirements are related to performing their duties and responsibilities, it is important to understand them. Table 2.2 summarises key duties and responsibilities for different roles.

No	Role	References	Duties and responsibilities
1.	Systems analyst	(Misic, 1996).	Defines requirements. identify and evaluate alternate solutions, coding, testing, training, conversion, making presentations and maintenance of the system
2.	Programmer	(Wikipedia contributors, 2007b; ACS, 2008)	Designing, modifying, testing and coding programs, unit testing, debugging, supporting users and solving problems with software.
3.	Architect	(Mills, 1985; The Open group, 2007a)	Designs the overall framework and externals of a system and guarantees its integrity
4.	Tester	(Patton, 2005)	Examining specifications and code, Prepare test plans and scenarios, carry out testing , document results, work with stakeholders (such as users and programmers) to resolve issues
5.	Designer	(Mills, 1985).	The designer designs the internals of the system taking over from the architect who designs the overall framework.
6.	Business Analyst	(Evans, 2004; Australian Computer Society, 2008)	Analyse, design and improve business practices and processes, evaluate business needs, and contribute to the design and development of a business solution, work closely with developers and end users to ensure technical compatibility and user satisfaction, ensure timelines and budgets are met, and oversee the implementation of a new system, write user manuals, provide or co-ordinate training to users of a new system Link business with IT.
6.	Project Manager	(Ives and Olson, 1981; Schwalbe, 2002)	Project managers have the overall responsibility to plan, execute and deliver the promised end-product or service to the customer. They carry out tasks such as defining scope, estimating work, developing task lists, assigning resources, reporting status and manage risks

Table 2.2. Responsibilities of roles

2.2.5 Shortfalls in the literature

A consolidation of shortfalls identified in the literature on knowledge and skill requirements is presented below.

2.2.5.1 Body of Knowledge such as SWEBOK, SFIA and TOGAF

Industry bodies, such as IEEE and The Open Group, have provided guidelines on the body of knowledge for the professional discipline. The guide to SE Body of Knowledge (SWEBOK) depth of treatment is geared towards defining and providing a foundation for curriculum development, certification and licensing (IEEE, 2004) (see Appendix 11.13 for summary details). Similarly, The TOGAF framework seeks to develop uniform skill definitions for architects for certification and licensing (The Open group, 2007b). The Skills Framework for the Information Age (SFIA) provides a common skills reference model to enable employers to carry out a range of HRM activities such as skill audit, planning, standardisation of job titles and functions (SFIA, 2008). These skill frameworks address the needs of the macro level of the industry and fail to consider[10] organisational specific knowledge such as application system, cultural and social knowledge. While these frameworks serve useful purposes, such as setting academic curriculum and licensing, these are not meant to specify the operational knowledge requirements to perform SE roles in enterprise context.

2.2.5.2 Managers or users opinion should not be the real guide to identify the knowledge and skill requirements

Many reported studies capture opinions from managers and users regarding the knowledge and skill needs for various roles in SE teams (Cheney and Lyons, 1980; Cheney, 1988; Watson et al., 1990a; Leitheiser, 1992; Monin and Dewe, 1994; Lee et al., 1995; Sawyer et al., 1998). Managers or users opinions are not a clear indicator of knowledge and skills required to perform various roles within the team. Manager's perspectives could be tainted since they are not performing the job to know what is required. Manager's opinions may help in understanding the knowledge and skill requirements from the point of view of human resource recruitment. While, users will be able to assess the output from SE teams they may have difficulty describing the knowledge by which the output was produced. There is already reported evidence for cognitive differences between system developers and users (Byrd et al., 1992). Further, Green (1989) found differences between systems analysts and users in their perceptions on skills and duties performed by systems analysts. Hence, only those who perform the job would be able to better articulate the actual knowledge and skill requirements to perform their job (to the extent possible, apart from tacit knowledge). Hence, perceptions on knowledge and skill requirements should be sourced directly from the individuals performing the job rather than associated stakeholders.

2.2.5.3 Weak emphasis on contextual knowledge types

Augier et al. (2001) quote Dilley and say that context is both constitutive of social action and the outcome of social action. Contextual knowledge refers to the knowledge of the contextual environment in which an individual performs the role. It is largely organisation specific. For example, the cultural knowledge of the organisation is a contextual knowledge that is required to perform a specific role. Context emerges through the individual interpretation of a situation including others and artefacts (Augier et al., 2001). In chapter 3, the contextual knowledge types are identified. The literature review found that only a weak emphasis has been given to

[10] except SWEBOK which includes process knowledge

contextual knowledge types such as 1) social network knowledge, 2) cultural knowledge, 3) configuration knowledge 4) process knowledge and 5) heuristics knowledge. Empirical research should be conducted to understand the importance of these knowledge types to performing various roles in SE.

2.2.5.4 Theory based knowledge requirements not available

There have been sound philosophical theoretical underpinnings for knowledge and action in disciplines such as Philosophy, Psychology and Mathematics (Popper, 1972; Anderson, 1982; Rogoff and Lave, 1984; Wertsch and Rogoff, 1984; Moore, 1985; Bandura, 1986; Anderson, 1993; McDonald, 2002; Tondl, 2007). The identified literature was analysed to see if any substantive theory underpinned the investigation and findings. This is reported in the summary of literature (see Appendix section 11.15; Table 11.5 - column C7). It is found that the reported research on knowledge and skill requirements to perform SE roles is not underpinned by sound fundamental theory on knowledge and action. It is important to understand the applicability of existing theories on knowledge and action to SE discipline.

2.2.6 Summary

The literature review shows that theories on knowledge and skill requirements, to perform various roles in SE, have been proposed and tested over the years in the literature. A comparative summary of these frameworks is presented in Table 2.1. A summary of the literature is presented in Appendix section 11.15.

The literature review demonstrates the need for 1) technical skills, 2) soft skills, 3) systems knowledge, and 4) business domain area knowledge to perform various SE roles. On the other hand, the literature review finds that there is a general lack of empirical testing of the need for the following knowledge types: 1) social network knowledge, 2) cultural knowledge, 3) configuration knowledge, 4) process knowledge and 5) heuristics knowledge. It is also found that environmental factors such as age, sex, organisational type (government vs private), organisational maturity can have an effect on the perceived knowledge and skill requirements to perform the job. Shortfalls exist in literature. It is found that existing knowledge/skill frameworks, such as SWEBOK, do not adequately capture the knowledge requirements to perform a job. Rather they focus on the macro-level industry needs. There is a general need to understand SE roles from a fundamental theoretical perspective on knowledge and action.

2.3 Performance

2.3.1 Introduction

WordNet defines performance, as the *"act of performing; of doing something successfully; using knowledge as distinguished from merely possessing it"* (Princeton University, 2007). In SE, individuals perform work to develop, enhance or maintain the business application software system by using knowledge, skills or abilities. The manner in which or the efficiency with which they are able to complete the SE task is referred to as the performance.

SE performance can be differentiated into two basic streams: Individual performance and Team performance. Team performance relates to the performance of a group of individuals, who perform different roles to produce outcomes (Henderson and Lee, 1992; Chung and Guinan, 1994; Sawyer and Guinan, 1998). However, this research investigates the relation between individual knowledge and individual performance.

2.3.2 Individual performance

Individual performance is the efficiency with which an individual is able to deliver the expected outcomes (Macquarie Online Dictionary, 2008). Curtis et al. (1988) provided evidence from a field study to the proposition that the thin spread of application domain knowledge and less social network knowledge affected performance. Zmud (1979) synthesised the findings of empirical investigations of the manner in which individual differences impact system development success and concluded that individual differences relating to cognitive behaviour have a direct impact on success. Rasch and Tosi (1992) developed an integrated model for performance based on Goal-setting model, Expectancy theory and Individual characteristics model and demonstrated that individual intellectual Knowledge, Skills and Abilities (KSA) has the strongest direct effect on perceived performance. Dispositional factors, such as 1) personality dimensions and 2) motivational factors were found to be related to job performance (Rasch and Tosi, 1992; Kantrowitz, 2005). Wade and Parent (2001) carried out a survey of Web masters and found that skill deficiencies in both organisational skills and technical skills lead to lower job performance. White and Leifer (1986) obtained perceptions from systems professionals on skills and performance and found that skills[11] such as technical skills, systems knowledge, business domain knowledge and soft skills have an impact on the overall performance. Thus, it has been demonstrated in the literature that skills and ability contribute to performance (Zmud, 1979; Rasch and Tosi, 1992; Wade and Parent, 2001). However, the existing research has not[12] drilled down the higher level factor of KSA into more detailed knowledge factors[13] . Very limited research exists on the relative contribution levels of these knowledge factors to performance. Further, any effort to understand knowledge and performance must take into account the job role.

2.3.3 Summary

In SE, performance refers to successful performance of the SE activities. There have been studies reported in the literature on the relation between KSA and performance. Almost all of these studies have used perceptible measures of performance and perceptible measures of skill/knowledge to understand this relationship.

2.4 Chapter Summary

This chapter presented the literature review on knowledge and skill requirements and their contribution to performance. The results provide a strong backing to the need for technical skills, soft skills, systems knowledge to perform various SE roles. The review finds that the contextual knowledge types such as social network knowledge, business domain knowledge, application system knowledge and process knowledge have been given little emphasis. It is also found that there is lack of sound frameworks that are based on fundamental theories on knowledge and action. While at a high level the research has found evidence between KSA and performance, the contribution of detailed knowledge factors to performance is not addressed in literature.

---

[11] White and Leifer (1986) defined skills as follows: 1) technical skills (including technical knowledge, being analytical and understanding business) and 2) process skills (including communication skills, leadership skills, decision making skills)

[12] White and Leifer (1986) skill divisions

[13] Similar to ones identified in literature review; such as soft skills or application system knowledge

3 Theoretical framework

3.1 Introduction

The literature review (Chapter 2) found that there is a general lack of knowledge and skill frameworks that are based on fundamental theories on knowledge and action. Further, it was found that the need for contextual knowledge types[14] to perform SE roles have not been given adequate emphasis. In order to understand the detailed knowledge factors, this chapter investigates the theoretical origins of expertise and postulates a fundamental theoretical proposition for knowledge and action in SE. Theories on knowledge, cognition, skill, competence, ability and expertise are drawn from the field of cognitive psychology. Cognitive and Social origin theories of knowledge and action are two schools of thought on expertise.

The organisation of this chapter is shown in Figure 3.2. Cognitive origin theories emphasise biological action of brain, while theories of social origins of knowledge emphasise the society around which individuals carry out action. Section 3.2 outlines the cognitive theories of knowledge specifically the Anderson's Adaptive Control of Thoughts (ACT) theory of expertise and how these theories can be applied to SE. This section identifies cognitive knowledge types in SE. Section 3.3 discusses the social origins of knowledge (theories on social, cultural and historical origins of knowledge). Analysis of how knowledge of social origins of need to perform to SE roles is then presented. Knowledge types of social origin in SE are then identified.

Reconciliation of social origin and cognitive origin theories has been carried out in literature. According to these theories, both prior cognitive knowledge and the knowledge of contextual or social circumstances shape the actions of experts. Section 3.4 discusses this reconciliation.
In SE, knowledge has a direct relationship to performance (White and Leifer, 1986; Rasch and Tosi, 1992; Wade and Parent, 2001). Using this relationship and applying reconciled cognitive and social origin theories, a basic theoretical model for SE (see Figure 3.1) is formulated. Section 3.5 presents the detailed theoretical framework for this research based on the reconciled model of Knowledge and Performance. Section 3.6 summarises the chapter by

[14] such as social network knowledge, business domain knowledge, application system knowledge and process knowledge

a brief review of the theoretical framework and its formulation and explaining how it is used in research design and presentation of results.

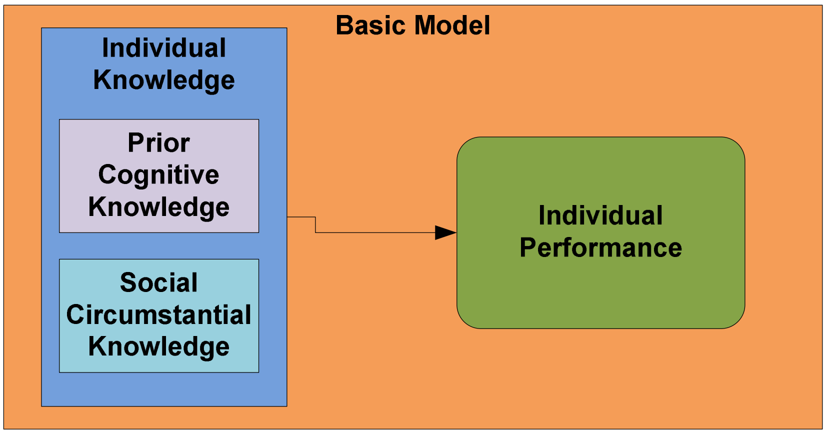

Figure 3.1. Basic theoretical model for this research

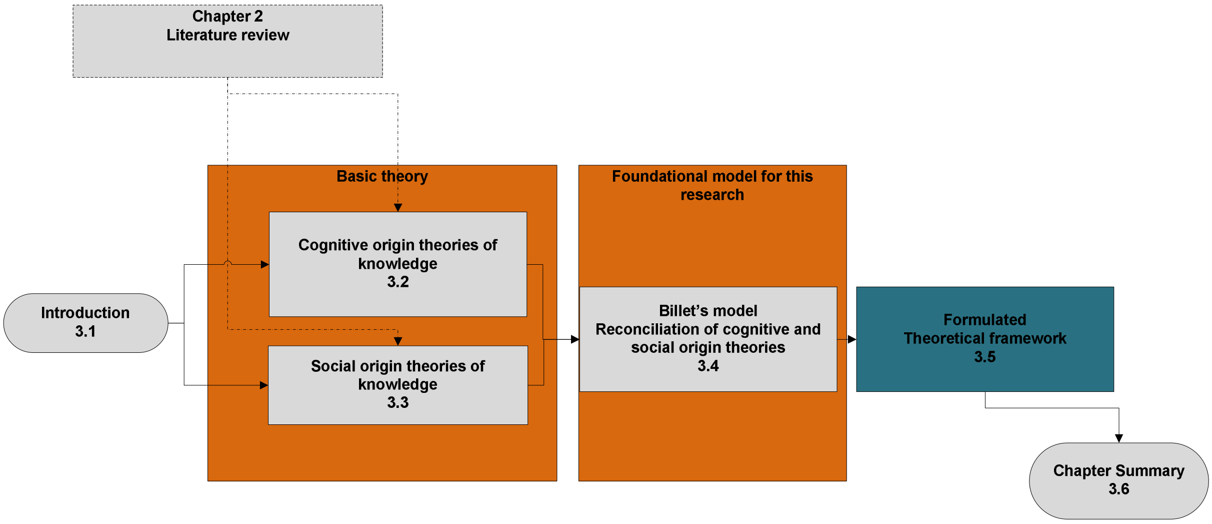

Figure 3.2. Flowchart: Organisation of Chapter 3

3.2 Cognitive origins of knowledge

SE roles require expertise. Expertise is highly skilled, competent performance in one or more task domain areas (Sternberg and Ben-Zeev, 2001). The gaining of expertise involves skill acquisition. Skill is defined as the ability that allows a goal to be achieved within some domain with increasing likelihood as a result of practice (Rosenbaum et al. 2001, cited in Eysenck and Keane, 2005).

Cognitive psychology explains study of thought and action by studying the internal workings of the brain (Medlin et al., 2004). Several theories on development of expertise have been proposed over the years in cognitive science (Eysenck and Keane, 2005). Anderson's ACT theory is one of the influential theories on Cognitive origins of Knowledge and Action. It should be noted that the theoretical understanding of cognitive representation and processing of knowledge is still rudimentary and is contested (Sternberg, 1994, cited in Robillard, 1999). However, the ACT theory is a useful way to analyse and explain the relation between knowledge and action, in the human mind.

3.2.1 ACT Theory

According to Anderson's ACT theory, there are three interconnected systems of cognitive structures[15] working together to perform an action as shown in Figure 3.3.

[15] representations in memory.

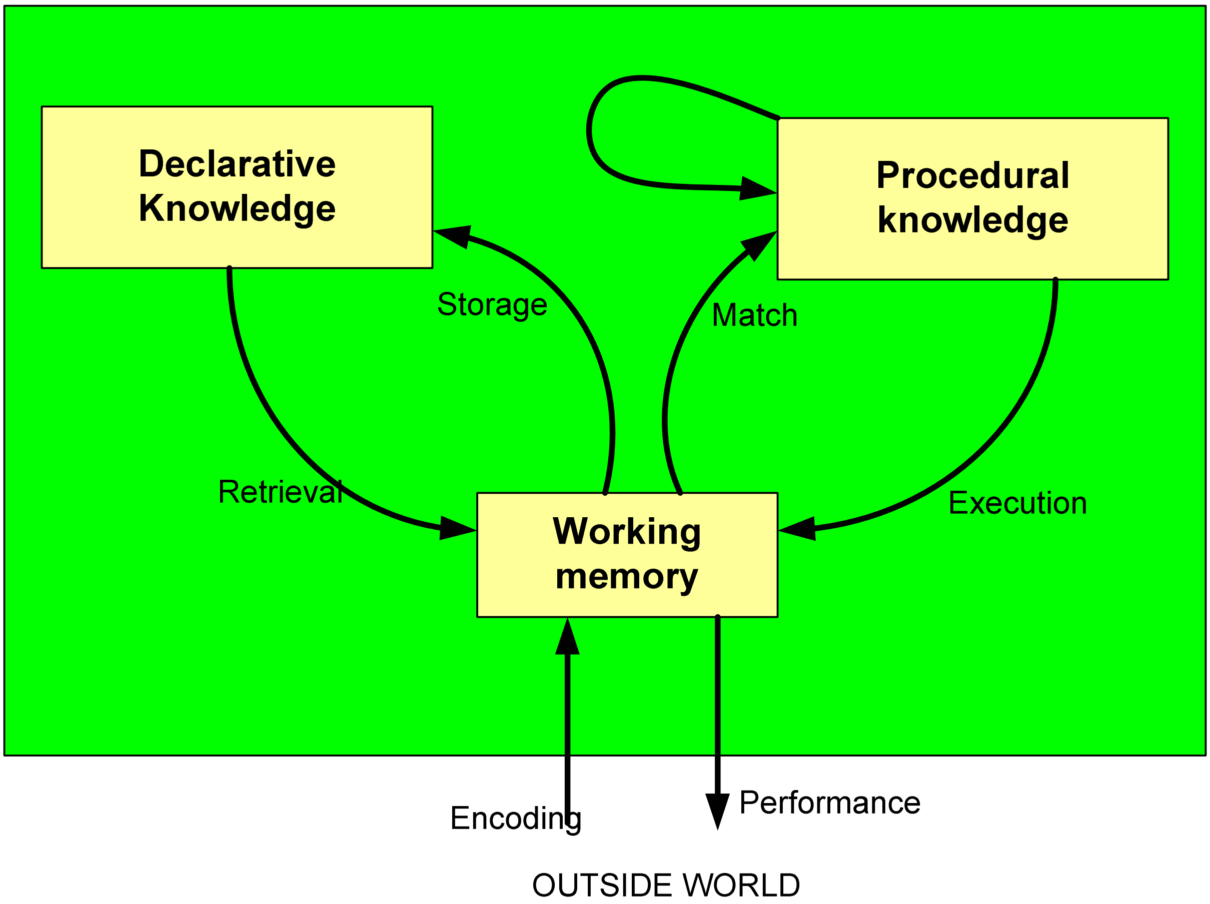

Figure 3.3. Anderson's ACT theory (Anderson, 1982; Anderson, 1993)

3.2.1.2 Declarative knowledge

Declarative knowledge comprises of encoded facts, information, assertions, concepts and propositions (Anderson, 1982; Anderson, 1993). This knowledge is represented as a semantic network of interconnected concepts (Eysenck and Keane, 2005). Experts posses simple factual knowledge to deep levels of domain specific knowledge (Billett, 1995; Awad and Ghasiri, 2003; Eysenck and Keane, 2005)

Declarative knowledge is made of two types of knowledge-: episodic and semantic knowledge.

Semantic knowledge is defined as a mental thesaurus, organised knowledge a person possesses about words and other verbal symbols, their meanings and referents, about relations among them, and about rules, formulas, algorithms for the manipulation of these symbols, concepts and relations (Eysenck and Keane, 2005).

Episodic knowledge relates to the encoding, storage and retrieval of knowledge about specific events or episodes occurring in a particular place at a particular time (Eysenck and Keane, 2005). Software development requires both semantic and episodic knowledge (Robillard, 1999).

Schemata are general knowledge structures or slots that represent the knowledge of the person's experiences, the relationship among objects, situations and sequences of events (Robillard, 1999; Medlin et al., 2004). Schemata are context dependent, rarely fully specified and contain assumed default values for variables (Robillard, 1999; Medlin et al., 2004).

3.2.1.3 Procedural knowledge

Anderson (1982) theorised that procedural knowledge enables individuals to perform skilful action. Procedural knowledge comprises techniques, skills to carry out tasks (Awad and Ghasiri, 2003) and includes problem solving skills (Anderson, 1993). Procedural knowledge is encoded in terms of production rules that are condition-action pairs (Eysenck and Keane, 2005). Procedural knowledge acquisition is based mainly on practice (Robillard, 1999; Awad and Ghasiri, 2003; Eysenck and Keane, 2005).

Individuals practice procedural knowledge by performing action when required. According to ACT theory, skill acquisition is due to a progressive shift from declarative knowledge to procedural knowledge (Anderson, 1982; Eysenck and Keane, 2005). This process is known as knowledge compilation and is made up of two sub-processes:

- Proceduralisation: Embedding factual knowledge into procedural rules.
- Composition: Collapsing collapses sequences of procedures into single procedures

Proceduralised rules are further refined during skill acquisition to produce more selective procedural rules based on processes such as generalisation and discrimination (Anderson, 1982).

3.2.1.4 Working memory

Working memory contains information that is currently active (Eysenck and Keane, 2005). Human mind has limited working memory. In order to circumvent the limitations of working memory plans are developed and implemented in human mind. Software engineers make plans to manage the working memory limitations (Robillard, 1999). Plans can be hierarchically structured or heuristic in nature.

3.2.1.5 Knowledge and Action

Each procedural rule has 1) a condition that specifies the circumstances under which the rule can apply and 2) an action that specifies what should be done when the rule is triggered (Anderson, 1982). According to Anderson's ACT theory, when an individual encounters conditions which match the stored procedural rules, then action occurs (Eysenck and Keane, 2005). This practice leads to the creation of new or specific procedural rules with associated conditions.

3.2.1.6 Dispositions

Dispositions are individual's tendencies to put their capabilities into action (Perkins et al., 1993). Dispositions include 1) inclinations, which may reflect motivation, habit, policy, or other factors; 2) sensitivity to occasion; and 3) abilities themselves (Perkins et al., 1993). Anderson's ACT theory did not consider the effect of dispositions on actions. However, Billett (1995) argues that dispositions need to be considered as a non-cognitive factor during cognitive action. In case of SE, Rasch and Tosi (1992) found that dispositional factors, such as achievement motivation, were a strong factor in determining a software developer's performance. Based on a survey in six large private organisations, Jiang et al. (1999) found that dispositions such as user orientations, political or technical orientation have a relation with perceived skill needs. Hall et al. (2008) suggested that motivational factors, such as 1) identification with the task, 2) good management, 3) rewards and incentives, 4) career path, and 5) recognition, as important could be related to project success.

The application of the ACT theory to SE is discussed in the following sub-section.

3.2.2 SE and its Cognitive origins

Numerous evidence is available in the literature on the cognitive-origins of knowledge for performing SE roles. Pennington et al. (1995) investigated the fundamental elements of skill and its acquisition in Object Oriented design in comparison to procedural design. Through experimental studies on designers, they found that the design process is cognitive in nature. As per Zmud's (1979) model of Management Information Systems (MIS) success the cognitive behaviour of individuals is considered as a significant contributor to MIS success. Robillard (1999) applied theories of cognitive origins of knowledge to explain software developer performance. Software designers use different types of cognitive planning approaches depending on the need and situation: 1) Systematic planning, 2) Opportunistic planning and heuristic planning, and 3) Serendipitous planning (Guindon and Curtis, 1988;

Guindon, 1990; Pennington et al., 1995; Sen, 1997; Robillard, 1999; Atman et al., 2007). It is found that expert programmers possess more useful mental models (LaToza et al., 2006) and are able to rapidly and effortless recognise problems, use higher-level plans than novices and are able to recall relevant knowledge (Mayer, 1992). This shows the cognitive nature of software development tasks. Rasch and Tosi's (1992) found that cognitive ability and dispositional factors contribute to individual performance in software development. The following sub-sections identify various knowledge types relating to cognitive origins theories.

3.2.2.1 Role of technical skills

According to the ACT theory, procedural knowledge comprises techniques, skills and the ability to secure goals (Anderson, 1982; Eysenck and Keane, 2005) and includes problem solving skills (Anderson, 1993). The technical skills that are required to perform various SE roles has been well-researched and reported in the literature (see section 2.2.2). Technical skills being specialised knowledge required to perform the role are part of procedural knowledge. A research study analysing developers work habits at Microsoft observed that developers relied on their internally stored technical knowledge by creating mental models of code (LaToza et al., 2006). Similar conclusions were drawn by Hutchins (1994, cited in Rosson, 1996) emphasising the importance of complex internal mental activities of programmers relating to technical skills.

A simulation of the cognitive knowledge structures, predominantly relating to technical skills, present in a developer's brain and how it changes over time has been reported in the literature (Hanakawa et al., 2002). Even though, this simulation is primitive in nature, as it is not grounded on any theory, it, none the less, illustrates the cognitive nature of technical skills. Based on these evidences from literature, technical skills can be classified to be a predominantly cognitive type skill.

3.2.2.2 Role of soft skills

Ehrlich (2004) defines soft skills as the skills needed to perform jobs where job requirements are defined in terms of expected outcomes, but the processes used to achieve the outcomes may vary widely. Being ways to secure goals, soft skills can be considered as a form of procedural knowledge. The cognitive nature of soft skills has been demonstrated in literature (Conrad, 1999 Gist and Stevens, 1998, cited in Kantrowitz, 2005). The need for soft skills to perform SE roles is well-researched and reported in the literature (see section 2.2.2.3). Hence, soft skills are identified as knowledge of cognitive origins.

3.2.2.3 Role of systems knowledge

Todd et al., (1995) consider problem-solving skills and analytical modelling skills as the key parts of systems knowledge. It is the understanding of the interplay of cause-and-effect relationships that is essential for problem solving (Beazley et al., 2002). Procedural knowledge includes problem solving skills (Anderson, 1993). Hence, systems knowledge is identified as knowledge of predominantly cognitive origins. The need for systems knowledge to perform various SE roles is well-researched and reported in literature (see section 2.2.2.6). This knowledge is a cognitive knowledge type required to perform the job.

3.2.2.4 Role of business domain area knowledge

Experts possess a comprehensive foundation of knowledge that is compiled over a long period of time, about their domain of expertise (Guindon, 1990; Ericsson and Lehmann, 1996; Cross, 2004; Atman et al., 2007). The business domain area in which the application system is rooted is an important cognitive knowledge for a person performing SE role. Expert systems literature has long recognised the cognitive nature of business domain knowledge and its need for experts (Premkumar, 1989; Valiris and Laios, 1995; Wagner et al., 2001). The need for business domain area knowledge has been well recognised in SE literature as well (see section 2.2.2.5). Based on these evidences, business domain area knowledge is identified as knowledge of cognitive origins.

3.2.2.5 Role of heuristics knowledge

Heuristic tasks are tasks for which experts use causal models or interrelated sets of principles or guidelines to decide what to do when (Reigeluth, 2002, cited in Lee and Reigeluth, 2003). In software design, heuristics are rules of thumb that reduce the complexity of a design problem. The importance of heuristics used by experts in SE was highlighted by Zmud (1979), early on, in his work on factors contributing to individual success. The systematic/heuristic dimension reflects whether an individual utilises abstract models and systematic processes in cognition or whether the approach taken is based more on experience, common sense and the practicalities of a situation (Zmud, 1979). Laboratory data collection and analysis of designer's thought processes has revealed that designers use different classes of heuristics such as the following (Guindon and Curtis, 1988):

- General problem solving heuristics such as add simplifying assumptions and consider temporarily a simple problem.
- General design heuristics (for example, trade-offs based on prioritisation)

- Software design heuristics (examples: looking at in an object oriented way, emphasising reliability based on risks)

Riel (1996) provides number of guidelines and rules of thumb for object oriented design. Hihn and Habib-agahi (1991) found that majority of technical staff estimating software costs use informal analogy and high level of partitioning of requirements to carry out cost estimation at Jet Propulsion Laboratory. Based on two empirical and controlled studies, Parsons and Saunders (2004) demonstrated that developers deliver artefacts that exhibit aspects of cognitive heuristics such as anchoring[16]. Heuristics such as standards from industry bodies such as CMMI and ISO (Software Engineering Institute, 2004; ISO-SPICE, 2008) and in-house organisational standards (Fitzgerald, 1997) are used in SE. The cognitive origin theories specify that experts in SE use heuristics to solve complex and ill-defined problems.(Robillard, 1999). Hence, heuristics knowledge is identified as knowledge of cognitive origins.

3.2.2.6 Cognitive origin theories: Identified knowledge types

The ACT theory postulates that experts possess three different types of knowledge: 1) Declarative, 2) Procedural and 3) Working knowledge. Declarative is based on facts or the type of knowledge referred to as 'what'. Procedural knowledge is the skill required to perform an outcome or referred to as 'how'. Individuals have a finite working memory. The human mind contains schemata, a knowledge structure, that contains relationships and links between underlying knowledge structures. Individuals need to plan to divide the main task into sub-goals to circumvent the problem of limited working memory. When an individual encounters a scenario, where the scenario conditions match a set of procedural rules, the rules are fired leading to action. Dispositions have an effect on the performance of the individual.

Cognitive knowledge includes compiled procedural knowledge and declarative knowledge that is gained over time. This knowledge that is based on cognitive origin theories is identified in this thesis. Such identification can help us formulate the theoretical framework. Based on

16 When people are given a problem for which there exists a correct solution and an initial estimate of its solution, they tend to provide final estimates close to the initial estimate when asked to solve the problem. This is termed as anchoring (Parsons and Saunders, 2004).

evidence collected in the literature survey (see 2.2) and in this section, following knowledge types are considered as cognitive knowledge types.

- Soft skills
- Business domain area knowledge
- Technical skills
- Systems knowledge
- Heuristic knowledge*

* - As we will see in section 3.3.3, heuristics knowledge has organisation specific parts as well.

These knowledge types have been identified based on cognitive origin theories and are used in the formulation of theoretical framework later on in this chapter (see section 3.5).

3.2.3 Summary

SE roles require experts, who possess high level of skills and knowledge. The concept of expertise has been explained based on the internal workings of the mind in the cognitive psychology theories. 1) Declarative knowledge, 2) procedural knowledge and 3) working memory work together to perform actions. Declarative knowledge refers to the 'what' and consists of factual knowledge. The procedural knowledge refers to the 'how' and consists of rules or condition-action pairs. The working memory is helpful to store current active knowledge during processing of actions. Experts possess detailed knowledge about their domain of expertise. An action is performed by the expert when the person encounters a condition that matches the rules in procedural knowledge. Plans are helpful to circumvent the issue of limited working memory. Such planning can be systematic or heuristic.

SE is a knowledge intensive activity. Much evidence exists in literature to support the proposition that cognitive processing of knowledge is significant to perform various roles. For instance, Zmud (1979) has shown that cognitive behaviour contributed to individual success. Robillard (1999) applied the cognitive psychology theories to software development. It has been shown that developers rely on the mental models of code (LaToza et al., 2006). Based on the literature review, the research has identified knowledge types: 1) soft skills, 2) business domain area knowledge, 3) technical skills, 4) systems knowledge and 5) heuristics knowledge as based on cognitive origin theories.

3.3 Social origins of knowledge

3.3.1 Introduction

In contrast to cognitive origins of knowledge, the social cognitive psychology theories emphasise cognition in the context of social interaction and behavioural adaptation (Barone et al., 1997). Sternberg (1989) wrote that theories on expertise need to be broadened to include socio-cultural factors such as personality and context. Social origin theories take the view that learners do not accumulate knowledge from the outside, but rather participate in activities that are distributed among the individuals, tools, and artefacts of a community (Mason, 2007).

This chapter presents the social origin theories of knowledge and action and its applicability to SE. Much evidence exists in literature to argue that SE is a social process. These evidences are presented in this section. Further, this section identifies the knowledge from social origins such as 1) cultural knowledge, 2) configuration knowledge (knowledge about the application software system), 3) process knowledge, 4) social network knowledge and 5) heuristics knowledge to performing SE roles.

3.3.2 Social origin theories

Every day ideas, representations and reasoning which govern mundane events are themselves the outcome of extensive social cognitive processes (Forgas, 1981). According to the social origin theories, the interdependence between the individual and the socio-historical environment is founded on developmental levels such as individual evolution, individual socio-cultural histories, continuous learning by individuals in particular problem situations (Rogoff and Lave, 1984). Lave and Wenger (1991), made a crucial step away from this individual perspective of knowledge, by proposing that learning is a process of participation in community of practice. Community of practice is a set of relations among persons, activity and world over time and in relation to other tangential and overlapping communities of practice (Lave and Wenger, 1991). An individual working in a community of practice becomes a practitioner and goes through life cycle phases such as newcomer, old-timer, whose identity is dependent on their changing knowledge or skill. Lave and Wenger (1991) further write that knowing is relational among practitioners, their practice, artefacts of the practice and social and political context of the community of practice. The social world of the community of practice undergoes change, reproduction and transformation, with cultural and political issues such as 1) mastery in communities, 2) problems of power, 3) access and 4) transparency. Learning occurs through participation in that community of practice.

Bandura (1986) put forward the Social Cognitive theory to explain the social foundations of thought and action. He says that 1) cognitive, 2) behaviour[17] and 3) other personal factors and environmental influences[18] operate in a triadic reciprocal deterministic way, influencing each other in a causal manner with differential contributions. This theory has been applied to develop a framework to elicit skills needed in software development (Downey and Power, 2007) and to understand dispositional factors such as person-organisational fit (Wingreen et al., 2003). While the social origin theories, proposed by Lave and Wenger (1991) focus on knowledge from social participation (or community of practice), the theory by Bandura (1986) emphasises interpersonal relations (Wenger, 1998). However, both theories emphasise the importance of reconciled social and cognitive origins of knowledge and action.

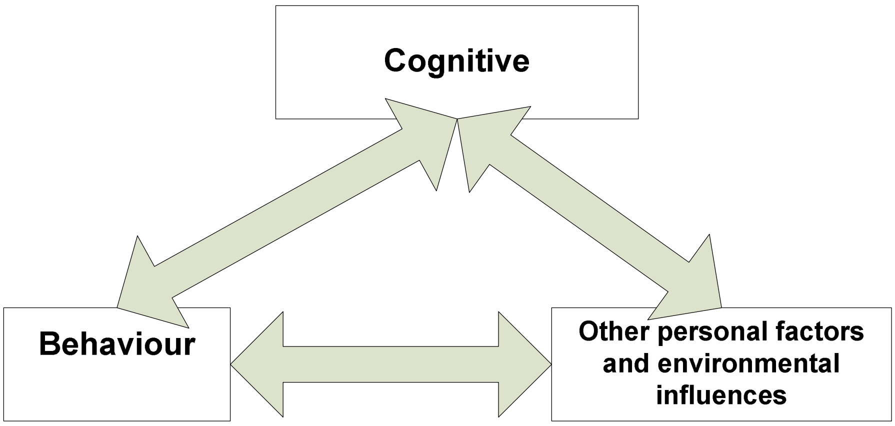

Figure 3.4. Bandura's (1986) Social Cognitive Theory: Triadic reciprocality

3.3.2.2 Summary

Knowledge has social origins. Learning is a process of participation in a community of practice. Expertise is relational to a community and culture of practices and has social origins. SE activities happen in a community of practice with a culture of practice. The next section presents the literature to support the view that SE is a social process and illustrates the

[17] Behaviour can be thought of as action or performance.

[18] Knowledge from social origins due to interpersonal relations

importance of knowledge from social origins such as cultural knowledge, configuration knowledge, process knowledge and social network knowledge to performing SE roles.

3.3.3 SE knowledge and its social origins

Gasson (1998) applied theories of social behaviour and social cognition on system design activity to arrive at models of software development processes. Based on an interpretive study, she identified the following key characteristics of software development design process:

- Situated in the organisational context and distributed amongst designers and other stakeholders the design was formulated.
- A community of sustained social practice allowed distributed cognition
- Design goals and boundaries emerged over time within the social contexts (example: external pressures for closures, early statement on design objectives and benefits circulated to senior management etc).
- Political nature of design activity – (example : influential decision-makers who wanted closures Vs designers who wanted to explore further)

Steen (2007) analysed qualitative data collected from interviews of 18 SE professionals and found that personal, practical knowledge, developed through experience, is important to perform various SE roles in addition to normative approaches such as methods, standards and metrics. Steen (2007) says in SE personal knowledge is a person's action oriented knowledge and is an integration of subjective experiences and rules of thumb, values, theories etc. mediated by culture and tradition. Such personal knowledge, when practice oriented, is referred to as practical knowledge. This work demonstrates the social origins of knowledge required to perform various roles.

Gasson (1999), further, proposed an integrated model for system design situated in an organisational context. She reported that design knowledge is distributed among a design team rather than shared inter-subjectively. Her work is a cornerstone in understanding the way systems design is carried out and aligns with theories of situated learning put forward by Lave and Wenger (1991). Gasson (1999) writes that knowledge is shared or distributed among software design team members and actionable in the context of the socio-cultural practices of the group. Zmud (1979) proposed a model which illustrates the manner in which individual differences are believed to influence MIS success. He divided individual differences into three classes: cognitive style, personality and demographic/situational variables. Cognition refers to the activities involved in attempts by individuals to resolve inconsistencies between an internalised conceptualisation of the environment and what is perceived to be actually transpiring in the environment (Zmud, 1979). Further, he says

> *"…it is becoming increasingly realised that cognitive behaviours are dependent on contextual, i.e., task and environmental factors as well as individual differences" (Zmud, 1979)*

SE is a team activity (Curtis et al., 1990; Rosson, 1996; Barthelmess and Anderson, 2002; Cubranic et al., 2004; He, 2004; Ye, 2006), which is collaboratively performed by various roles of persons and tools (Saeki, 1995). Architects, developers, business analysts, project managers and testers, other individuals performing roles and other stakeholders need to work together to produce an outcome in an enterprise context. The importance of social technical challenges to coordination and cooperation in SE including issues such as processes, tools, communication, negotiation have been highlighted in literature (Yamauchi et al., 2006; de Souza et al., 2007; Herbsleb, 2007; Panjer et al., 2008). Sawyer et al. (1996) carried out an analysis of electronic team room activity and found that the shared screen of the team screen acted as a social buffer and enabled social interaction. Sawyer et al. (1996) demonstrated the relation between team room use and improved product quality. Sawyer et al. (1996) observed that the electronic team room became a mechanism to address antagonism such as intra-group conflicts. Similar, tools that enable communication, collaboration and cooperation in software development have been found to be useful to improve performance (Saeki, 1995).

Chung and Guinan (1994) found that two contextual factors, 1) team size and 2) professional experience of team members moderated the relationship between participation and performance. The importance of contextual knowledge to application development has been observed through ethnographic studies of software integration projects (Yamauchi et al., 2006). Guinan et al. (1998) empirically demonstrated that group processes are important predictors of team performance in requirement determination phase. These reported research work are evidence to the appropriation of knowledge through collaboration and antagonism in the social origins of knowledge.

Rajeswari and Anantharaman (2003) carried out a survey to measure sources of pressure among software professionals. Using factor analysis of data they arrived at 10 social factors that contributed to stress, as follows: 1) fear of obsolescence, 2) individual team interactions, 3) client interactions, 4) work family interface, 5) role overhead, 6) work culture, 7) technical constraints, 8) family support towards career, 9) workload and 10) technical risk propensity. Based on a survey of managers from 120 organisations, Dyba (2005) found that dispositional factors and knowledge management initiatives contributed to success of Software Process Improvement (SPI) initiatives.

The importance of social factors such as conversations, satisfice[19] in design decisions has been found to be relevant in design decisions in a qualitative study (Zannier and Maurer, 2007). It has been found, based on a survey, that system analysts perceive their role as a socio-technical function (Lerouge et al., 2005). Rosson (1996) says in real life software development projects, the major focus is on team dynamics. Guinan and Bostrom (1986) proposed communication based models to improve productivity of developers and users. Team dynamics and communication relate to the social nature of SE. McDonald (2001) evaluated and compared different expertise locator systems in software development and found that expertise was relational to community of practice and demonstrated the importance of local knowledge to experts. These empirical` research results show us that SE is indeed a social process with diverse range of social issues including culture, personal life histories and dispositional factors. Hence, the literature provides clear evidence that SE is a social process. The following sub-sections identify various types of knowledge types relating to social origins theories.

3.3.3.1 Role of process knowledge and cultural knowledge

Being a social process, culture and processes are a complex part of the overall SE process. The importance of processes for SE has been well understood and emphasised (Humphrey, 1995; Wasserman, 1996). Barthelmess and Anderson (2002), modelled SE process by applying the concepts of activity theory and complemented Gasson's (1999) work. They primarily viewed software development as a collaborative activity where the environments can significantly influence the collaborative nature of software development work. Sawyer and Guinan (1998) found that social processes, such as the level of informal coordination and communication, the ability to resolve intra-group conflicts and the degree of supportiveness

19 Satisfice refers to a decision being satisfactory rather than being optimal.

among the team members, can account for 25 percent of the variations in performance, illustrating the importance of process and culture. Klappholz et al. (2003) developed an instrument to measure the attitude, assess the knowledge and ability to apply the software development process. They note that successful software development requires expertise in both the software technology and software development process (Klappholz et al., 2003). Robey and Newman (1996) proposed a social process model for systems development and tested its predictive capabilities in case studies. Robey and Newman (1996), explaining the process model, provide scenarios such as experiences of user domination, attempts to persuade then coerce users and hostility between users and systems development staff. These episodes and scenarios illustrate the influence of culture and processes in systems development. Dyba (2005) based on an empirical study, identifies Knowledge and its management as key contributors to Software Process Improvement (SPI) success. The importance of process knowledge to SPI initiatives and the need to manage them have been well-recognised in the literature (Arent and Nrbjerg, 2000; Mathiassen and Pourkomeylian, 2003; Rodriguez et al., 2004; Alagarsamy et al., 2007; Santos et al., 2007).

Janz and Licker (1986) came up with three key dimensions of culture to assess the relationships between systems departments and their clients. These dimensions illustrate the importance of culture to SE roles, in working with clients to accomplish outcomes:

- Concern : Showing that systems division cares about its products and image and tries to understand its customers to get its products approved or used
- Contact : The frequency with which systems division actually meets with or consults with its clients
- Control : The degree to which systems division relies upon “the book” to accomplish its work or get it approved or used

Cultural influences, such as work behaviours, organisational culture, communication style, issues in global SE teams, and their effect on productivity have been studied in literature (Borchers, 2003; Herbsleb, 2007; Huang and Trauth, 2007; Olson, 2007; Aurum et al., 2008). For instance, Borchers (2003) identified three key indices[20] of culture that impact the way the geographically-spread teams function. Olson (2007) documented the problems facing software development teams that are located in multiple geographic locations spread

20 power distance, uncertainty avoidance and individualism.

remotely. He identified issues due to cultural differences such as Team composition (members of the team and what motivates them) and Team work (in which the activity progresses including predilection for planning, decision making processes and taking responsibility). The collective culture of the team is found to contribute to knowledge transfer within team, in a study conducted amongst university students (Joshi and Sarker, 2006). Thus, in order for some one to perform well in a SE context, one needs to have an understanding of the processes involved and the prevalent culture in that context. Hence, cultural knowledge and process knowledge are significant to SE roles.

3.3.3.2 Role of social network knowledge

Perry et al. (2001) based on an observational case study of parallel changes in software development found that developers and project managers communicate and coordinate informally and need to understand what each other is doing showing the importance of social network knowledge. Sheng et al. (2003) demonstrated that socio-cultural factors impact what IT professionals believe about their IT skills. Nakayama and Sutcliffe (2001) proposed an IT skills management framework based on a socio-cultural perspective addressing temporal concerns such as the rapid change in technologies, mentoring, coordination and need for political skills needed to handle social settings. Rus and Linwall (2002) write that capturing knowledge needs and *who knows what* is a key driver for KM in SE. Radha Mookerjee (2005) found that social network knowledge related issues such as collaboration between departments and organisational structure issues were common problems in enterprise application software maintenance. In reporting an observational study of six managers, Ives and Olson (1981) found that relationships are more important for managers. The importance of cooperation and coordination to SE, including social and technical dependencies between individuals has been well understood, since many tools have been designed to alleviate the difficulties associated with them (McDonald, 2001; de Souza et al., 2007; Herbsleb, 2007; Panjer et al., 2008).

Busch et al. (2001) carried out a study to understand the information flows within systems organisation. They found strong evidence for the social nature of tacit knowledge and that transfer of tacit knowledge is overwhelmingly through human to human contact. They further showed that even a single individual could affect tacit knowledge information flows within the systems domain. Such key members, denoted as brokers, have been found to hold pockets of knowledge and have an impact on information flow (Marczak et al., 2008). Herbsleb

(2007) writes that given the reduced level of communication in global software development efforts, the problem of understanding what other project members are doing, in order to coordinate effectively with them, is much more difficult. Bassellier and Benbasat (2004) demonstrated, based on survey results, the importance of partnerships with business clients to performance, thus illustrating the importance of social network knowledge. Based on case studies of three organisations, Mathiassen and Vogelsang (2005) demonstrated the dynamic nature of social networks and their role in software methodology adoption. Thus, these existing literatures provide us the evidence for the need for social network knowledge to perform various SE roles.

3.3.3.3 Role of configuration knowledge

The functional and technical knowledge of the application software system is referred to as the configuration knowledge in this thesis. This knowledge is specific to the organisation and the application system and hence is classified as part of the social circumstantial knowledge. Sawyer and Guinan (1998) and Rilling (2002) found that the application system knowledge was organisation specific and a critical asset for developers due to issues such as attrition and new members taking to learn the application system. This thesis already reports evidence (see section 2.2.2.4) from existing literature on the need for configuration knowledge to perform various roles in software development (Zmud, 1983; Curtis et al., 1988; Herbsleb and Kuwana, 1993; Walz et al., 1993; LaToza et al., 2006). The core focus of an enterprise SE team is to develop, modify or maintain application system. Based on these, the configuration knowledge is identified as a predominantly contextual in nature.

3.3.3.4 Role of heuristics knowledge

Based on ACT theory, it was argued (in section 3.2.2) that heuristic knowledge can be identified from cognitive origin theories. However, as we will see here, this knowledge type has organisational specific nature as well. Zmud's (1983) categorised heuristic knowledge items as relating to policies of the organisation. Chan (2003) found that heuristics such as company specific policies and rules are important to performing various roles in Enterprise System development. Tacit heuristic knowledge, that has a soft dimension, such as guidelines, short cuts, underlying assumptions, macro-institutional factors, standards and policies, are largely organisation specific (Selamat and Choudrie, 2004; Guzman and Wilson, 2005). Hence, heuristic knowledge can also be identified to be knowledge from social origin theories.

3.3.3.5 Social origin theories: Identified knowledge types

Based on the above discussion of social origins of knowledge for SE, it is found that any proposed theoretical framework on knowledge and skill requirements for SE need to consider the following types of organisational contextual knowledge that is related to the social circumstances.

- Process knowledge
- Cultural knowledge
- Social network knowledge
- Configuration knowledge
- Heuristic knowledge*

* - Heuristic knowledge has both cognitive and circumstantial origins

These knowledge types identified from social origin theories are used in the formulation of theoretical framework later on in this chapter (see section 3.5).

3.3.4 Summary

The social origin theories emphasise the importance of socially derived knowledge to performing a job. Such knowledge can be evolved or derived historically, based on a community of practice and are dependent on individual's personal life histories and their socio-cultural backgrounds. Individuals learn new knowledge when they are involved in problem solving situations. Expertise is relational and is based on a socio-cultural setting.

Much evidence exists in literature that SE is a social process. For instance, Zmud (1979) recognised the importance of context to cognitive behaviours and Sawyer and Guinan (1998) demonstrated the relation between social processes and performance in software development. Process knowledge, cultural knowledge, social network knowledge, configuration knowledge and heuristics knowledge are organisation specific and are based on social circumstances.

3.4 Reconciled Cognitive and Socio-cultural theories

3.4.1 Introduction

The importance of brain and its function is emphasised in cognitive psychology literature while social origin theories emphasise the cultural and social origins of knowledge such as the community of practice. The reconciliation between both the cognitive and social origin theories has been proposed so that the processes of thinking and acting can be understood more deeply (Mason, 2007). This section presents the literature on reconciled theories. The reconciled theory is then applied to SE (in the next section) to develop the theoretical framework for this research.

3.4.2 Reconciliation in Educational Psychology literature

Educational Psychology literature reports work on reconciliation between the cognitive and social origin theories. Billett (1995; 1996; 2001) and Packer and Goioechea (2000) reconciled both literatures and developed models of knowledge and action for an expert.

Billett (1995; 1996) investigated the representations of knowledge and problem-solving strategies of practitioners or experts and found that they are not only shaped by prior cognitive knowledge, but also are shaped by social circumstances in which they conduct their practice. He found evidence to the model of knowledge, its sources and its change (see Figure 3.5) by applying and testing the model in a vocational setting (hair dressing) . Experts before they are involved in problem situation or goal directed activity possess prior knowledge gained through experience, training and education. The social circumstantial knowledge is gained through involvement in the goal directed activity. Through involvement in problem solving activity individuals gain new knowledge.

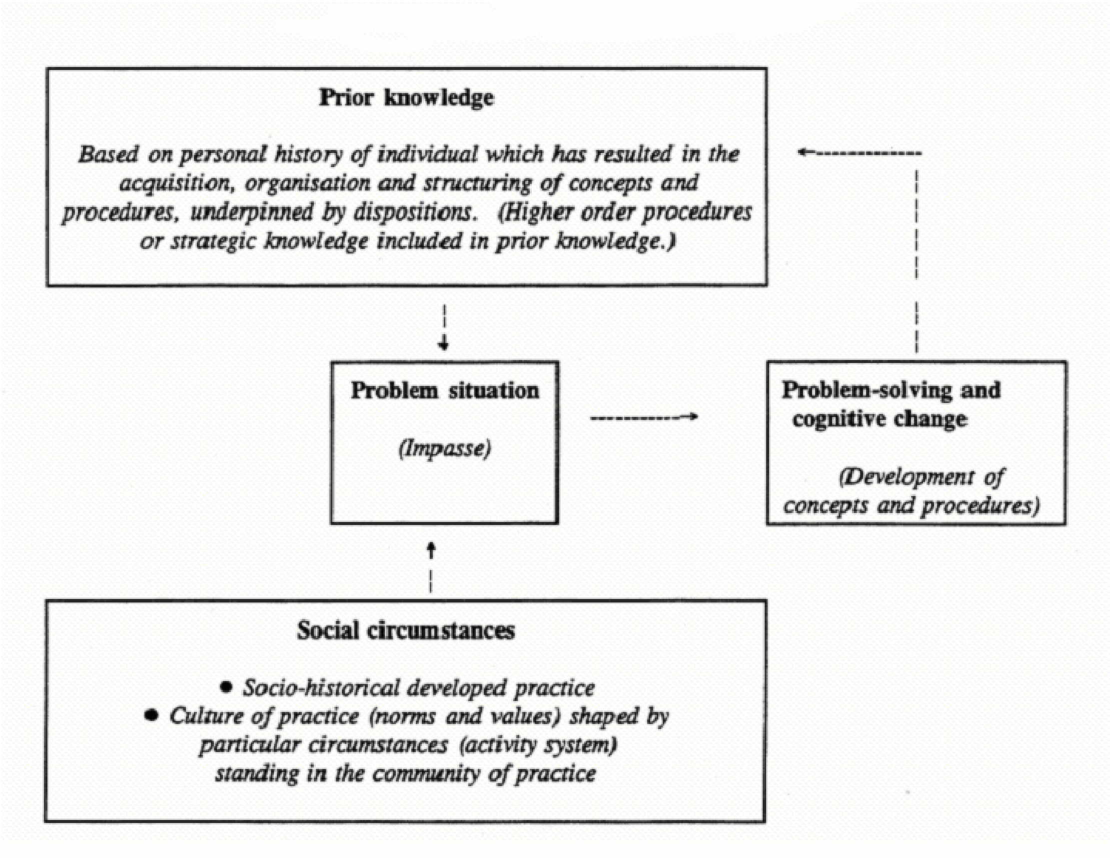

Figure 3.5. Billett's (1995) Model of Knowledge and Performance

Packer and Goioechea (2000) proposed that learning involves 1) becoming a member of a community, 2) cognitive construction of knowledge at various levels of expertise as a participant, and 3) also taking a stand on the culture of one's community. They analysed the process of learning by children in classrooms and found that learning involves levels that include individual cognitive knowledge and the social context of the schooling community. They conclude that acquiring knowledge and expertise always entails participation in relationship and community. This results in transformation both of the person and of the social world.

3.4.3 Theory of internal and external cognition

Shariq (1999) proposed a theory in the KM arena to explain the knowledge transfer process. This theory aligns with reconciled theories on cognitive and social origins of knowledge. According to Shariq (1999), human cognition is enacted interactively and dynamically. He defines internal cognition as the accessed knowing of tacit, codified abstractions and artefacts by an enactor in a particular situation. The internal cognition as described by Shariq (1999), relate to an individual's personal cognitive structures as postulated by the cognitive origin theories and focuses on personal mental representations (Mason, 2007). External cognition is the embodiment of the natural and the artifactual environment with which the enactor is interacting in that particular situation including organisational culture and social networks (Shariq, 1999) and can be thought of as knowledge of context. Hence, external cognition is knowledge of social origins. External cognition or the knowledge of the contextual environment has been found to be significance in knowledge transfer (Augier et al., 2001) and in systems organisations (Mathiassen and Pourkomeylian, 2003; Guzman and Wilson, 2005; Mathiassen and Vogelsang, 2005b). Shariq (1999) posits that the interaction is between internal and external cognition and in order to enact, both of these must be accessible within the human mind. The theory proposed by Shariq (1999) is in alignment with Zmud's (1979) model of MIS success, where in cognition refers to the activities involved in attempts by individuals to resolve inconsistencies between an internalised conceptualisation of the environment and what is perceived to be actually transpiring in the environment (Zmud, 1979). Thus, the reconciliation of cognitive and social origin theories is proposed and debated in KM literature and aligns with SE literature.

3.4.4 Competency and Operational knowledge

The concept of competency relates to one's ability to perform a specific job. Blanton et al. (1998) defined Information Technology Professional Competency (ITPC) as the degree to which one possesses expert knowledge and skills, and exhibits behaviours that are effective in performing the design, development and consulting duties relevant to the contemporary Information Technology (IT) profession (Blanton et al., 1998). They divided ITPC into five factors: technical knowledge, organisational knowledge, competency behaviours, employability and professional knowledge. Bassellier and Benbasat (2004) defined business competence as the set of business and interpersonal knowledge and skills possessed by IT professionals that enable them to understand the business domain, speak the language of business and interact with their business partners. Business competence included organisation specific competencies, social networks, interpersonal and leadership skills.

Since, the word *knowledge* has many connotations, due to its varying characteristics such as its tacit nature, Sveiby (1997) suggests that the word *competence* be used to denote an individual's ability to perform a job. Sveiby (1997) says competence consists of five mutually dependent elements : 1) Explicit knowledge 2) Skill, 3) Experience, 4) Value judgements and 5) Social network.

All the three definitions of competence (Sveiby, 1997; Blanton et al., 1998; Bassellier and Benbasat, 2004) illustrate the importance of contextual organisation specific knowledge as well as cognitive skills. However, the Human Resource Management (HRM) literature defines competency in a slightly different manner as the underlying characteristic of a person, that results in effective and/or superior performance in a job (Boyatzis, 1982, cited in Lindgren et al., 2004). Competence, in HRM literature, includes motivation, disposition, self-image, values, moral standards, norms of social behaviour and traits as well as communication, general reasoning and learning capabilities (Lindgren et al., 2004). Hence, the definition of competence in HRM literature appears to have a slightly broader scope than the definition put forth by Sveiby (1997) with the key difference in the inclusion of dispositional factors.

A similar terminology that is close to the definition of competence by Sveiby (1997) has been developed by Beazley et al. (2002). Operational knowledge is thought of as the tacit knowledge required to perform well in a given position. Operational knowledge consists of critical data, information, formal processes, informal processes, skill sets, applied experience, relationships, competencies, beliefs, values and wisdom that create the domain of understanding that enables employees to excel at the tasks they undertake (Beazley et al., 2002).

The job-specific operational knowledge includes the following: (Beazley et al., 2002)

- Cognitive Knowledge: Content knowledge that includes job-specific data and information and their sources.
- Skills knowledge: The skills and training necessary to perform well in the position
- Systems knowledge: An understanding of the interplay of cause-and-effect relationships that is essential for sound decision making
- Social network knowledge: An understanding of crucial social relationships.
- Process and procedural knowledge: Knowledge of formal and informal organisational processes and procedures.

- Heuristics knowledge: Knowledge of shortcuts for accomplishing tasks, rules of thumb for decision making and quick fixes.
- Cultural knowledge: Knowledge of organisational norms, values, roles and standards of conduct that govern interaction with colleagues and other stakeholders.

The term operational knowledge as defined by Beazley et al. (2002) and the term competence as defined by Sveiby (1997) represent the same underlying concept that is the knowledge, skills and abilities required to perform well in a given position. Both competence and operational knowledge include prior cognitive and social circumstantial knowledge in their definitions. Hence, the concepts of competence and operational knowledge align with the reconciled theories of cognitive and social origins of knowledge and action (Billett, 1995; Billett, 1996; Shariq, 1999; Packer and Goicoechea, 2000; Augier et al., 2001).

3.4.5 Application to SE context

SE is an intense knowledge driven activity and is similar to professional practices such as medicine and accounting. Much evidence exists in literature (see section 3.2.2) that SE roles require knowledge that has cognitive origins. Section 3.3.3 presents the literature on the social origins of SE knowledge needs. Hence, the skills and knowledge required to perform SE both require contextual and prior cognitive skills and knowledge to perform the job.

The reconciled theories of cognitive and social origins of knowledge have been tested in vocational practice and education. These reconciled theories are well-founded on scholarly scientific schools of thought on Psychology that have long history. This thesis has provided an exhaustive literature survey and evidence that justifies the cognitive and social origins of knowledge for SE (see sections 3.2.2 and 3.3.3). Hence, this research adapts and applies this reconciled theory of cognitive and social origins of knowledge as the foundation for this research, in answering the research questions that have been posed.

3.4.6 Summary

The reconciled theories of cognitive origins of knowledge and social origins of knowledge explain the relationship between knowledge and action in goal directed activity. Individuals make use of both prior cognitive knowledge and social circumstantial knowledge to resolve problems. Such participation in goal directed activity results in increased knowledge. This thesis has provided an exhaustive literature survey and evidence that justifies the cognitive and social origins of knowledge for SE. Hence, this research adapts and applies this reconciled theory of cognitive and social origins of knowledge as the foundation for this research, in answering the research questions that have been posed.

3.5 A theoretical framework for this research

3.5.1 Introduction

The aim of formulating a theoretical framework is to identify the factors and their relationships so that it can be tested and provide answers to the originally posed research questions. The reconciled cognitive and social origin theories (see section 3.3) provide a framework to explain the relation between knowledge and action to perform SE roles. The Billett's (1995) model shown in Figure 3.5 is a useful representation of the reconciliation and will be used as the basis for further discussion in this thesis. Based on this model the theoretical framework is formulated and presented for this research.

3.5.2 Theoretical framework

This thesis aimed at identifying the knowledge factors needed to perform various SE roles. It further aimed to identify the knowledge factors that contribute to an improved performance. It has already been demonstrated in earlier studies that knowledge has a direct relationship to increased performance (White and Leifer, 1986; Curtis et al., 1988; Rasch and Tosi, 1992; Wade and Parent, 2001). Applying, Billett's (1995) model to this theoretical statement, a basic theoretical model for this research is formulated (see Figure 3.6). This model includes both prior cognitive knowledge and social circumstantial knowledge. Detailed knowledge factors can now be incorporated into this theoretical framework based on the literature review.

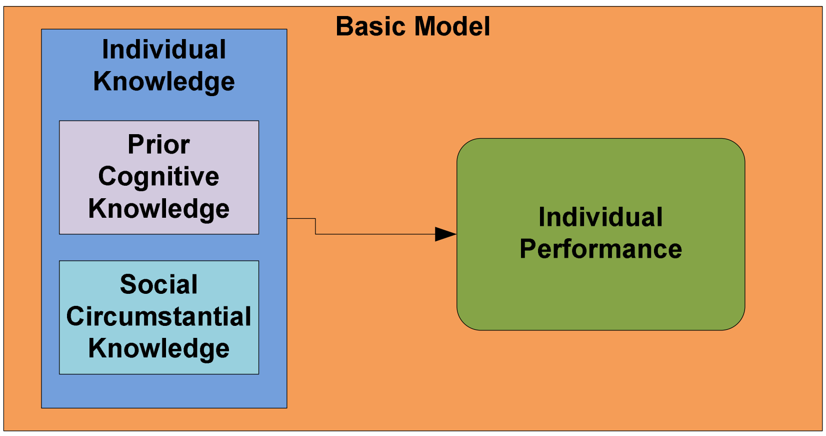

Figure 3.6. Basic theoretical model for this research

As discussed in section 3.4.4, Beazley et al. (2002) framework of operational knowledge types closely aligned with Billett's (1995) model. This framework was adapted. Modifications were carried out to Beazley et al.'s (2002) list of operational knowledge types. These are explained below.

3.5.2.2 Changes to skills knowledge

Skills knowledge is a broad term that denotes many different types of skills. This term needs to be customised specifically towards SE. The researcher made use of the classification of skills knowledge by Bailey and Stefaniak (2001). According to them, skills knowledge for SE consists of three parts:

- Technical skills,
- Business concept skills and
- Soft skills.

The following changes were carried out to the above factors:

- Technical skills are changed to Technique skills – Generally Technical skills correspond to technology skills and specify skills such as .NET or Java (example: (Sawyer et al., 1998; Liu et al., 2003). Even in the work by Bailey and Stefaniak (2001) technical skills represented information technology skills. Since, the research is attempting to find the knowledge requirements for roles other than programmers as well the researcher changed the term to 'technique skills'. Technique skill is defined as the specific ability that is required to accomplish the tasks relevant to their role. For example, a programmer will require programming skills in a specific technology to accomplish his or her role. A project manager would require project management skills as the technique skills.
- Business concept skill to business domain area knowledge: Literature review indicated that words such as application domain area (Herbsleb and Kuwana, 1993) and business and functional skills (Lee et al., 1995) are used in literature. In order to clearly communicate the importance of business and domain area, business domain area knowledge was chosen.

In summary, based on the above discussion, the following detailed factors are introduced instead of skills knowledge.

- Technique skills – Specific abilities required to accomplish tasks required to perform the role.
- Soft skills.
- Business domain area knowledge

3.5.2.3 Changes to cognitive knowledge

Beazley et al. (2002) define cognitive knowledge as the content specific knowledge that is very specific to the job. Since all individual knowledge is cognitive in nature, the term cognitive knowledge could be misleading. For SE, the core focus and specific content is the business application software system. Hence, this knowledge refers to the application system knowledge. After due consideration of various terminologies the words *Configuration knowledge* was chosen since it better represented the term. Macquarie dictionary (2008) defines the word configuration as the relative disposition of the parts or elements of a thing. The grey literature shows us that configuration refers to a system comprised of a specified set of components or elements of specified versions (Wikipedia contributors, 2007a). Hence, configuration knowledge is defined as the knowledge of the application software system relevant to the SE team, including its functional and technical knowledge.

Based on the above discussion, the following factor is introduced instead of cognitive knowledge in Beazley et al. (2002) classification.

- Configuration knowledge: Knowledge of the application software system including the functional and technical knowledge.

3.5.2.4 A theoretical framework for the research

Applying, Beazley et al.'s (2002) concept of operational knowledge types to basic framework (shown in Figure 3.6) a final theoretical framework is formulated for this research and is shown in Figure 3.7.

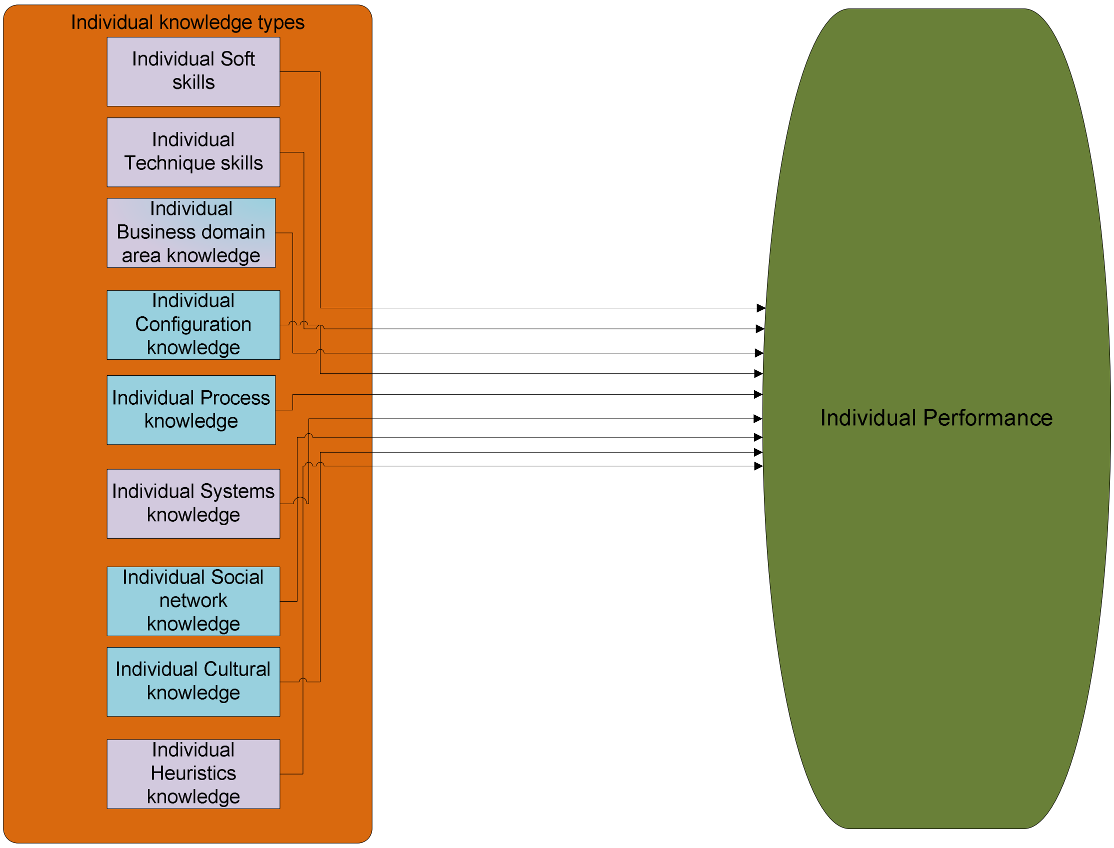

Figure 3.7. Theoretical framework : Knowledge required to perform SE roles

As per the model shown in Figure 3.7, nine different operational knowledge types contribute to performance for different SE roles. These are 1) configuration knowledge (functional and technical knowledge of application system), 2) social network knowledge, 3) process and procedural knowledge, 4) systems knowledge, 5) business domain area knowledge, 6) soft skills, 7) technique skills, 8) cultural and 9) heuristics knowledge. This theoretical framework is the foundation on which the research is built and provides the logical base for testing (Sekaran, 2003). This research set out to answer two research questions.

1. What are the relevant types of knowledge required to perform various roles in an enterprise application SE context?
2. What are the contributing factors to improvement in individual performance from a knowledge point of view?

This theoretical framework has identified a list of knowledge factors that are required to perform SE roles. As per this model, the individual's knowledge of different factors contributes to them performing their role. Any improvements in knowledge of these different knowledge types will result in an increased performance. Each of the different knowledge types identified above can contribute differently to the improvement of performance. It is envisaged that by using this model, that is founded on theories from Education, Cognitive and Social Psychologies, KM and SE, it is possible to answer both research questions.

3.5.3 Summary

Reconciled cognitive and social origin theories of knowledge and action (Billett, 1995; Billett, 1996; Sveiby, 1997; Shariq, 1999; Packer and Goicoechea, 2000; Augier et al., 2001; Mason, 2007) provide us with a useful basis for explaining SE knowledge and action. Using Billett's (1995) model (see diagram in Figure 3.5) as the basis, a preliminary theoretical model has been formulated. According to this preliminary model, both prior cognitive knowledge and social circumstantial knowledge are required to perform various roles in SE. This preliminary model has been expanded based on Beazley et al.'s (2002) work on operational knowledge. Beazley et al.'s (2002) framework incorporates both prior cognitive and social circumstantial knowledge to perform the job. By applying this framework a final theoretical framework encompassing nine types of knowledge is formulated. The formulated theoretical framework is the foundation for this research and drives the research including in design, in operationalisation of factors into items and in analysis.

3.6 Chapter Summary

SE roles require experts, who possess high level skills and knowledge. The concept of expertise has been explained based on the internal workings of the mind in the cognitive psychology theories. Experts possess detailed knowledge of their domain. SE is a heavily knowledge intensive activity. Much evidence exists in literature to support the notion that cognitive processing of knowledge is important to perform various roles (see section 3.2.2). Based on review of literature, the thesis has identified knowledge types such as 1) soft skills, 2) business domain area knowledge, 3) technical skills, 4) systems knowledge and 5) heuristics knowledge as knowledge types based on cognitive origin theories.

Individuals working in SE are social beings and are part of the society. These individuals work together as a team to build application software in enterprises. The social origin of knowledge theories emphasise the importance of socially derived knowledge in action. Much evidence exists in literature that SE is a social process (see section 3.3.3). This thesis identifies 1) process knowledge, 2) cultural knowledge, 3) social network knowledge, 4) heuristics knowledge and 5) configuration knowledge as based on social circumstances.

Reconciliation between cognitive and social origin theories (Billett, 1995; Billett, 1996; Sveiby, 1997; Shariq, 1999; Packer and Goicoechea, 2000; Augier et al., 2001; Mason, 2007) explains the relationship between knowledge and action in goal oriented activity. Individuals make use of both prior cognitive knowledge and social circumstantial knowledge to resolve problems. Using Billett's (1995) model as the representative basis for reconciled theories(see Figure 3.5) and adapting the framework of operational knowledge by Beazley et al. (2002), the theoretical framework is posited. According to this final theoretical framework, nine types of knowledge contribute to performing different roles in SE. These knowledge types are 1) configuration knowledge (functional and technical knowledge of application system), 2) social network knowledge, 3) process and procedural knowledge, 4) systems knowledge, 5) business domain area knowledge, 6) soft skills, 7) technique skills, 8) cultural and 9) heuristics knowledge. The formulated theoretical framework drives the research further including research design, operationalisation and analysis.

4 Research design

4.1 Introduction

This research is driven by a desire to build theory, by providing empirical evidence to answer the following research objectives set forth in this thesis.

- Identify the knowledge requirements to perform SE roles
- Identify the knowledge factors that contribute to individual performance improvement in SE

This thesis has so far formulated a theoretical framework based on a synthesis of cognitive and social origin theories of knowledge and action. Evidence needs to be collected to substantiate and test the proposed theoretical framework so that conclusions can be drawn. Two research methods are used to address the research questions:

- A case study research method: This single-embedded case study of a SE team captures the real life perceptions and experience of participants in an enterprise application software system context.
- A survey (referred to as general-survey) of individuals performing various SE roles.

The organisation of this chapter is shown in Figure 4.1. Following this introduction, section 4.2 presents the high level research design and the justification for choosing the research methods. Section 4.3 presents the operationalisation of the factors identified in the theoretical framework into operational items that can be measured. The research makes use of operational items from existing literature wherever possible. Where such items are not available or not suitable for this research, items are operationalised in consultation with experts from academia and practice.

Section 4.4 discusses the relevance of survey research method and describes the detailed research design including instrument design, sampling and data collection methodology, and analysis plan. Section 4.5 describes the case study protocol including the data collection procedures, instruments and analysis plan. Section 4.6 summarises the research design chapter.

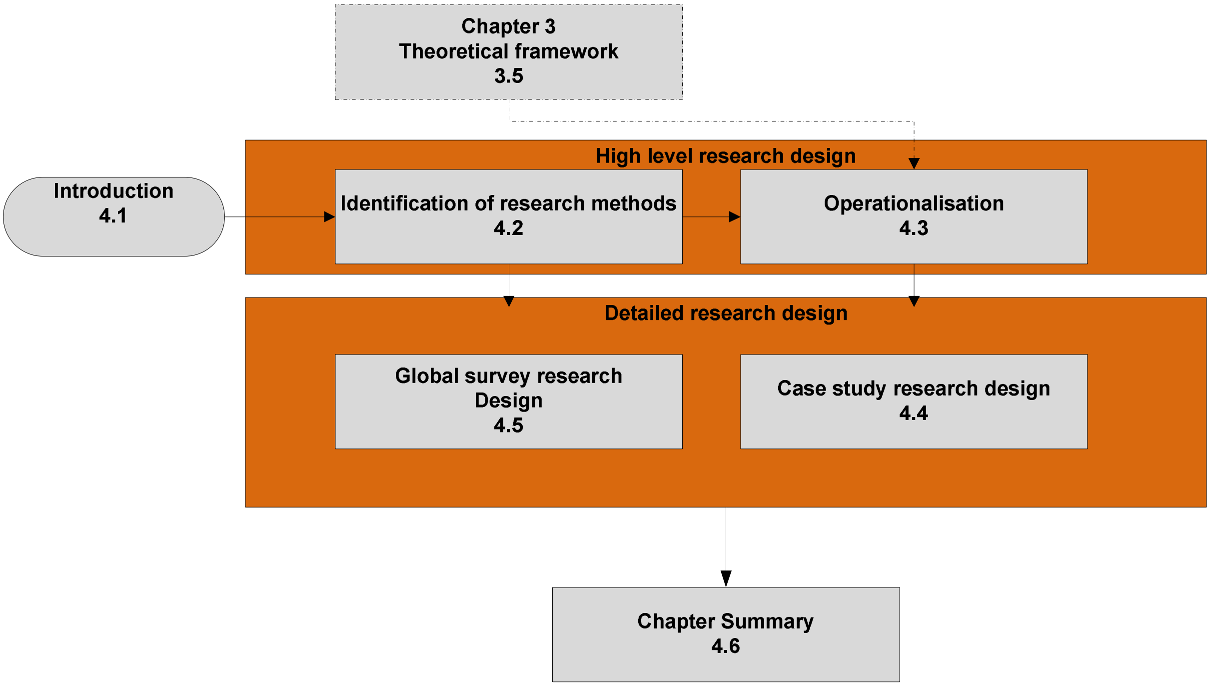

Figure 4.1. Flowchart: Organisation of Chapter 4

4.2 Identification of research methods

4.2.1 Introduction

This section identifies the research methods for answering the research questions. The research methods were chosen based on consideration of factors such as the suitability of the research methods, philosophical positions, feasibility of conducting the research, researcher's expertise, and availability of resources. The research project context analysis is presented in section 4.2.2. The chosen research methods and the reasoning behind choosing them are presented in section 4.2.3.

4.2.2 Research project context analysis

The research reported here is a conscious premeditated inquiry (Goldhor, 1972, cited in Williamson et al., 2000) to address the research questions, in the context of SE, to increase the knowledge of human kind. A theoretical framework has been formulated to address the formulated research questions. A positivistic empirical strategy is chosen to provide increased validity (Hamilton and Ives, 1982) and use is common place in the SE/IS discipline (Alavi and Carlson, 1992).

Individual knowledge is cognitive in nature (Davenport and Prusak, 1998). Perception studies have traditionally been used when 1) cognitive and mental behaviour, 2) skills, 3) knowledge and 4) abilities are involved in the operational variables (Cheney and Lyons, 1980; Cheney, 1988; Leitheiser, 1992; Rasch and Tosi, 1992; Lee et al., 1995; Nansi, 1998; Wade and Parent, 2001; Kannan and Akhilesh, 2002). Further, such perception studies have been used for theory building purposes in SE and IS disciplines (Examples: (Kannan and Akhilesh, 2002; Leon, 2002; Roberts and Gregor, 2005; Yaobin and Tao, 2007)). Perception studies are practical to conduct and hence are chosen for this research. The research question presented here is in the nature of finding the associated variables rather than to establish a cause-and-effect relationship. Further, the research is intended to be carried out with minimal or no interference to the work environments.

This research is focussed on deriving new knowledge through theory building for application to specific practical problem in practice. Hence, this research is categorised as applied research (Williamson et al., 2000). Theory is a body of knowledge that aids in understanding, explaining, and predicting the things we see around us, as well as providing a basis for action in the real world (Gregor, 2006). Gregor (2006), based on the purpose the theory serves for knowledge building, classifies theories into five categories[21]. The theory for analysing and describing state ‘what is’. On the other hand, theories for predicting say ‘what will be’.

This research principally seeks to identify and classify the knowledge and skills required to perform SE roles. Descriptive theories, such as classifications, are valuable when little is known about the phenomenon (Gregor, 2006). It has been found from literature that shortfalls on knowledge and skill requirements exist including lack of emphasis on contextual knowledge types (see section 2.2.5). Hence, this research is principally classified as theory for analysing and describing.

Surveys either try to discover facts about a population (Buckingham and Saunders, 2004) and offer a suitable avenue for investigation for theory building purposes, including descriptive theories (Tanner, 2000). Numerous research papers have been published before to answer similar research questions in SE/IS arena using survey research (Nelson, 1991; Lee et al., 1995; Nansi and Bennett, 1998). Further, the researcher is able to gain access to a pool of SE professionals for conducting the survey. Hence, the survey research method has been chosen, to understand the relationships between various variables, for this research.

The research questions posed are practice-based. Hence a descriptive analysis of a real life scenario of SE project team is important in providing a context in answering the questions. A case study research investigation[22] provides this opportunity to investigate the research questions in the field. Lethbridge et al. (2005) say that it is essential to conduct field studies in SE to study real practitioners solving real problems. Case studies can be used for theory building and to provide evidence to substantiate proposed theories (Sekaran, 2003). Case study methods have been widely used in SE/IS research (Benbasat et al., 1980; Sawyer et al., 1998; Suan et al., 2000; Sarker and Lee, 2003). Further, the researcher was able to gain access

---

21 The types of theories are 1) Analysis, 2) Explanation, 3) Prediction, 4) Explain and Predict, and 5) Design

22 A variant of case study is the field study research (Nansi and Bennet, 1998; Darke and Shanks, 2000). The difference between the two research methods is often difficult to discern, however, and is really a matter of degree (Darke and Shanks, 2000). This research adheres to the concepts proposed by Yin (2003) and uses the terminology of case study research method.

from an organisation to carry out case study research. Hence, case study research method was chosen.

4.2.3 Chosen research design

Figure 4.2 shows the research design. The research is driven by the formulated theoretical framework (see Chapter 3). Two research methods are used to address the research questions:

- A survey (referred to as general-survey) of individuals performing various SE roles.
- A case study research method: This single-embedded case study of a SE team captures the real life perceptions and experience of participants in an enterprise and application software system context.

Both the general-survey and case study research capture data on the knowledge and skill requirements to perform various roles in SE as well as the knowledge factors that contribute to improved performance. The case study research provides the opportunity to gather information in a specific organisational context. Both the case study and general-survey investigation make use of the formulated theoretical framework developed in this research (chapter 3). The operationalisation of theoretical constructs is common across both studies with the case study containing additional items that are based on the enterprise context. By capturing the perceived knowledge requirements for performing SE roles and factors influencing performance improvements it is possible to collect evidence, analyse and conclude answers to the posed research questions.

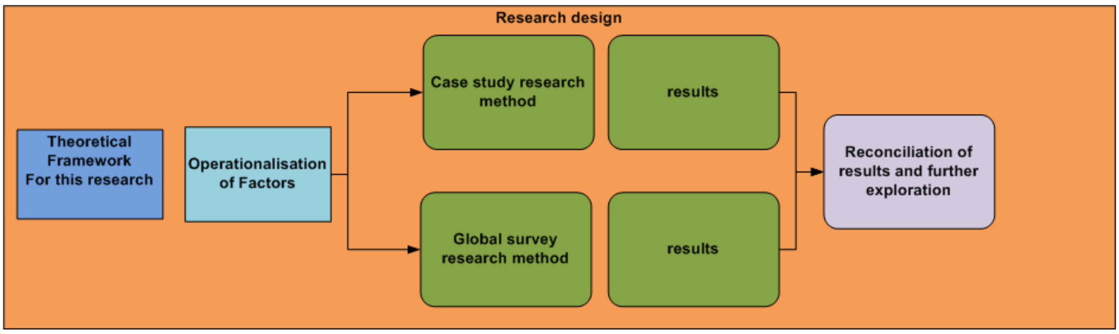

Figure 4.2. Research design

Using a combination of methods to address research questions is common in SE/IS research (Kaplan and Duchon, 1988). This research reconciles the findings from both case study and general-survey in order to answer the research questions. Both case study and general-survey research investigations are based on the developed theoretical framework and factors. The operationalisation of these factors into operational items (operational variables) is also common across both methods except where the enterprise contextual information can be adopted. Hence, the results of survey and case study method are reconciled to answer the research questions.

4.2.4 Summary

Selection of research methods is based on an analysis of the context in which the research project is based. This research is aimed at finding the knowledge requirements and knowledge factors that contribute to improved performance. The scientific research reported in this thesis is aimed at improving theory as well as providing important application to SE practice. The research questions posed not only aim to understand descriptive information but goes to explore and understand relationships between knowledge and performance. The unit of analysis for the research questions is at the individual level. Survey method offers the ability to develop descriptive theories. The insights and understanding of actors from real life situation is best provided by case studies. Hence, a combination study is chosen. It is important to note that the perceptions of individuals on knowledge and performance are captured by this research. A reconciliation of the results of general-survey research method and the case study method is made.

4.3 Operationalisation of theoretical variables

4.3.1 Introduction

So far in the thesis, the theoretical framework and methods with which the research will be conducted have been defined. This section discusses the details of how each of the variables is operationalised into items. Both case study and general-survey research make use of the same theoretical framework and operationalisation. However, the operationalisation of some of the variables into operational items in case study makes use of the organisational contextual information. This difference is explained in section 4.3.2. The roles that are considered for this research are outlined in section 4.3.3. The operationalisation of theoretical variables into various operational items is presented in section 4.3.4. The operationalisation of variables to assess the contribution to improved performance is presented in section 4.3.5.

4.3.2 Difference between the case study and general-survey

It is important to clearly distinguish the difference between the case study and general-survey research from an operationalisation of variables perspective. Case study is carried out in a real software project team in an organisational context. The purpose of the case study research is to present the view from this context. Hence, some of the knowledge and skills will reflect the real names of teams, individuals, and reflect on the culture in the organisation. For example, when we need to operationalise social network knowledge, the exact details of social networks operating for that software project team are used in the questionnaire. On the other hand, with general-survey research, questions ought to be generic and cannot be specific to the context. This difference between case study and general-survey is explained in relevant places in this section.

4.3.3 Roles considered for this research

The two research questions aim to identify the knowledge requirements and their contribution to performance required for various SE roles. Hence, it is important to identify the roles. The following roles will be the primary roles for which this research will identify the operational knowledge requirements: 1) Programmer, 2) Programmer-analyst, 3) Business analyst[23] and 4) Project Manager. Testers, architects and designers were also included as peripheral roles.

In order to reduce potential issues due to role ambiguity and overlap of role (Downey, 2006; Downey and Power, 2007), the responsibilities of the role are also captured in the research instrument and summarised as part of the results. Further, the research instrument allows for selection of multiple roles and allows entry of other not-predefined roles (see question A.6 in Appendix 11.4).

4.3.4 Operationalisation of Knowledge variables

As discussed in Chapter 3, the theoretical framework identifies nine knowledge variables for SE roles. This section operationalises these knowledge variables into operational items. Measuring knowledge poses challenges. Measurement of knowledge is complex (Freeman, 2001) due to its dynamic (Sveiby, 1997), tacit nature[24] (Busch et al., 2001). Knowledge requirements to perform a job can be measured through perceptions of individuals or through observations of behaviours. As not all knowledge will be manifested through behaviours (Sage and Rouse, 1999), measuring actions or behaviours won't provide a full picture of knowledge requirements for SE. Measurements of direct knowledge of individuals using instruments (such as the certification for domain knowledge (Freeman, 2001)) will not elucidate what knowledge was required to exhibit behaviours. Further, all the possible behaviours that are exhibited by individuals performing a SE role cannot be tracked.

Perception studies to measure knowledge provide a practical way to measure the knowledge requirements to perform a job. Perception based studies have disadvantages in terms of being perceptions as opposed to measuring reality. Such studies also cannot measure tacit knowledge. However, perception studies have largely been used in SE and IS disciplines to measure knowledge and skill requirements (McCubbray, 1988; Nelson, 1991; Leitheiser, 1992; Nord and Nord, 1997; Nansi and Bennett, 1998). Despite these disadvantages, due to their practical nature, this research makes use of perception studies. The following sub-

---

[23] In the last few years, the role title of business analyst appears to have over taken the title of systems analyst. (Reedman, 2004). Hence, business analyst is used instead of systems analyst.

[24] With articulable and inarticulable parts of tacit knowledge (Busch et al., 2001)

sections operationalise knowledge variables into operational items using perceived knowledge requirements to perform SE roles.

4.3.4.1 Soft skills

Ehrlich (2004) defines soft skills as the skills needed to perform jobs where job requirements are defined in terms of expected outcomes, but the processes used to achieve the outcomes may vary widely. The literature generally refers to communication skills, leadership skills, interpersonal skills as soft skills (Orsted, 2000; Bailey and Stefaniak, 2001). The literature review (see section 2.2.2.3) showed that various types of soft skills such as communication skills, interpersonal skills and leadership skills have been found to be important to perform various roles in SE. Bailey and Stefaniak's (2001) operationalisation of soft skill items were adapted for this research with modifications. From its definition, the operational item *'ability to visualise or conceptualise'* relates to a systems knowledge item[25] and hence was not included in the final list. Another operational item *'ability to apply knowledge'* is too generic[26] in nature and was not considered as a fair operational item for a soft skill. Two of the cultural knowledge items that directly referred to culture were removed from the list. The following items remain from the list provided by Bailey and Stefaniak (2001) and were the chosen as the list of soft skills.

- Team work skills (long term)
- Time management skills
- Adaptability to new technologies & languages
- Verbal communication skills
- Inter-team communication skills
- Investigative skills
- Organisational skills
- Ability to give and receive constructive criticism
- Ability to multitask
- Leadership skills
- Negotiation skills

---

[25] Systems knowledge includes visualisation of cause and effects (Beazley et al., 2002 p.28), analytical modelling and problem solving (Todd et al. 1995).

[26] This item can include any type of knowledge and its application and hence it was considered that this may not represent any specific soft skill.

- Interpersonal skills
- Stress management skills
- General writing skills
- Idea initiation skills
- Listening skills
- Technical writing skills

The operational items for soft skills within general-survey questionnaire and case study are identical.

4.3.4.2 Technique skills

Technique skill is defined as the specific ability required to accomplish the tasks directly relevant to their role. For example, a programmer will require programming skills in a specific technology to accomplish his or her role. A project manager would require project management skills as the technique skills. This research considers the following key roles (see section 4.3.3): 1) Programmer, 2) Programmer-analyst, 3) Business analyst and 4) Project Manager.

The core set of IT skills required to perform the role of a programmer and programmer-analyst, such as software skills, have been identified in the literature. The researcher operationalised the variable into various technique skill items, based on the role and responsibilities definition of the role.

For the business analyst, the core technique skill is the business analysis skill. For the project manager, the core technique skill is the project management skill. For a tester, the core technique skill is testing skill. For the programmer the core technique skills are the software skills and hardware skills. Programmer-analyst's role is a combination of the roles of a programmer and that of a business analyst. Hence, no additional technique skill item was required for programmer-analyst.

The technique skill was operationalised into the following skills.

- Software skills required to develop the application software
- Hardware skills required to develop the application software
- Business analysis skills
- Testing skills
- Project management skills

- Knowledge of application software architectures

The operational items for technique skill within general-survey questionnaire and case study will remain the same.

4.3.4.3 Business domain area knowledge

The business domain area knowledge relates to knowledge of the domain area in which the application system and the organisation is operating in. The survey participants may work on SE applications that relate to business domains such as insurance, banking, and airline. Survey research can seldom deal with such contextual information (Barribeau et al., 2005).

Bailey and Stefaniak (2001) operationalised business concept skills as a single item. Similarly, Sawyer et al., (1998) operationalised general business skills as a single item that related to business domain area knowledge[27]. Hence, business domain area knowledge is operationalised into one item as follows:

- Business domain area knowledge relating to the application system

4.3.4.4 Configuration knowledge in SE

Configuration knowledge is the technical and functional knowledge of the application software system. Software is defined to include written programs or procedures or rules and associated documentation pertaining to the operation of a computer system (WordNet 2.0 Search, 2004). The list of configuration knowledge items can be obtained by collecting inventory list of all software items present in the application software system(s) that the SE team is working on. In order to operationalise the configuration knowledge, the concepts from function point analysis method were adopted.

Function point method is one of the most popular estimating methods for application software. This method identifies five basic functions in application SE. These are as follows (Humphrey, 1995):

- Inputs: Screens or forms for user input.
- Outputs: Screens or reports that application software produces as output.
- Inquiries : View only screens that allow operators to interrogate an application and obtain assistance or information

[27] Excludes organisational contextual knowledge, which fall under other knowledge areas.

- Data files: Collection of data stored separately in storage.
- Interfaces: Interface with external applications.

This research makes use of these functions to operationalise configuration knowledge. The classification from function point analysis is used as a cue to operationalise the configuration knowledge variable. By adopting the functions and based on discussions with academics and experts in the software field, the researcher operationalised the configuration knowledge variable into the following operational items:

- Knowledge of user interfaces within the application system
- Knowledge of the interfaces of the application system with other systems
- Knowledge of program functionality within the system
- Knowledge of any reusable component functionality within the application system
- Knowledge of database structures of the application system
- Knowledge of the overall architecture of the application
- Knowledge of the software development environment at the software project

For the case study research project, based on the analysis of the application system artefacts within the case study project team, the actual names of software items (examples: user interfaces, components, programs, external interfaces to the system) identified in the application software system will be specified in the questionnaire. Software object[28] is defined as an aggregation of software, such as a computer program or database, that satisfies an end use function and is designated for specification, qualification testing, interfacing, configuration management, or other purposes (Thawyer, 2004). These software objects are selected based on trade-offs among software function, size, host or target computers, developer, support strategy, reuse plans, criticality, interface considerations, the need to be separately documented and controlled, and other variables. While operationalising configuration knowledge, it is important to note that the separation of application system into software objects is largely discretionary to the organisation and is contextual depending on various trade-offs mentioned by Thawyer (2004).

[28] referred to as software item by Thawyer (2004). In this thesis, software item is referred as software object to reduce confusion with operational item.

4.3.4.5 Social network knowledge

Knowledge flows along existing pathways in organisations need to be understood to leverage them (Prusak undated, cited in Anklam, 2003). These pathways refer to the social networks and relationships within the organisation. Curtis et al. (1988) found that a software engineer would normally communicate most frequently with team members, slightly less frequently with other teams on the project, much less often with corporate groups and except for rare cases, very infrequently with external groups (Curtis et al., 1988). Based on this model the operational items for social network knowledge were developed. Since, social network knowledge is about '*understanding who knows what'* (Rus and Lindvall, 2002; Cross and Parker, 2004) it was used in the operational items. The operational items for social network knowledge were formulated considering the above literature, as follows:

- Understanding of 'who knows what? ' within software project team
- Understanding of 'who knows what? ' within organisation but outside software project team
- Professional network of people outside the organisation

Social network analysis (SNA) is the mapping and measuring of relationships and flows between people, groups, organizations, computers, web sites, and other information/knowledge processing entities (Krebs, 2008). Such a full scale SNA can throw light into the full social networks that are relevant to the case study team. SNA has special ethical implications (Borgatti and Molina, 2003) and the researcher had not obtained (and won't be able to) obtain permission for doing the same within the case study SE team. Hence, this research did not carry out a social network analysis of the environment. It is difficult to identify the networks outside the organisation during the project profile phase. Hence, the case study project-questionnaire will retain the following two operational items as is:

- Understanding of 'who knows what? ' within software project team
- Professional network of people outside the organisation

However, the intra-organisational social network knowledge, based on an analysis of the environment within the organisation around the case study project team, will be replaced with the actual names of teams with which the case study project team interacts in the questionnaire.

4.3.4.6 Cultural knowledge

Culture includes knowledge, experience, beliefs, values, attitudes, meanings and hierarchies amongst a group of people (Samovar and Porter, 1995). Culture is a knowledge asset for the organisation and is embedded in social processes, institutional practices, traditions and technologies, artefacts (Boisot, 1998).

Janz and Licker (1986) operationalised culture using three variables concern, contact and control. Olson (2007) operationalised cultural differences using variables such as team composition and team work. Borchers (2003) made use of indices such as power distance, uncertainty avoidance and individualism. SE activity happens in a community of practice within an organisational context. Hence, this research is primarily interested in operationalising culture in an organisational context. The organisational behaviour literature refers to culture as the sum of the organisational member's views, attitudes and beliefs (Robbins et al., 2001). Robbins et al. (2001) summarised a seven-item description of organisational culture as shown below:

- Innovation and risk taking: The degree to which employees are encouraged to be innovative and take risks
- Attention to detail: The degree to which employees are expected to exhibit precision, analysis and attention to detail
- Outcome orientation: The degree to which management focuses on results or outcomes, rather than on the techniques or processes used to achieve these outcomes
- People orientation: The degree to which management decisions take into consideration the effect of outcomes on people within the organisation.
- Team orientation: The degree to which work activities are organised around teams rather than individuals.
- Aggressiveness: The degree to which people are aggressive and competitive rather than easygoing.
- Stability: The degree to which organisational activities emphasise maintaining the status quo in contrast to growth.

The above operational items for organisational culture have been adopted for this research.

Literature review suggests that knowledge sharing culture in organisations is an essential precursor for effective knowledge management (Davenport and Prusak, 1998; Dixon, 2000;

Rumizen, 2002; Awad and Ghasiri, 2003; Cross and Parker, 2004). The knowledge of '*who knows what*' and the associated knowledge sharing culture of the individual are very much interrelated. (Rus and Lindvall, 2002). Hence, a separate cultural knowledge item was introduced to specify the knowledge sharing culture within the organisation.

For the case study research, based on the analysis of documentation and other artefacts available in the organisation on culture, additional cultural knowledge operational items are added to the list of items presented above.

4.3.4.7 Heuristics knowledge

Heuristics refers to a useful shortcut, an approximation or a rule of thumb for guiding search (Gigerenzer and Todd, 1999). In real life every one of us uses heuristics in some form or other. Similarly, the tasks of a software developer also involve heuristic and procedural aspects. Lee and Reigeluth (2003) classify heuristics knowledge as guidelines, explanatory models, descriptive models and meta-cognitive decision rules. Due to its tacit nature (Busch et al., 2001), heuristics knowledge is very difficult to operationalise and measure (Freeman, 2001). Also, the operational items should be simple enough so that participants can understand and provide a response. For example, a questionnaire item such as meta-cognitive decision rules would lead the participant with no clear understanding. Beazley et al. (2002) defined heuristics knowledge as the knowledge of shortcuts for accomplishing tasks, rules of thumb for decision making, and quick fixes that have come to constitute best practices for the position. Based on this definition, two key operational items for heuristics were captured: rules of thumb and shortcuts. The item *guidelines* has been adopted from the work by Lee and Reigeluth (2003). Standards are an important part of SE processes. External standards (such as IEEE 828-1983, CMMI, and ISO-9000), organisational standards and guidelines (examples: user interface guidelines (Smith and Mosier, 1986)) and security standards (Defence Signals Directorate, 2008) are adapted and used in a specific SE environment (Sommerville, 2001) for various SDLC phases. Hence, the operational items 1) standards and 2) guidelines have been added to the heuristic knowledge items. Thus the following items have been used to operationalise heuristics knowledge.

- Knowledge of shortcuts to accomplish various tasks
- Knowledge of guidelines within the organisation
- Knowledge of standards within the organisation

- Knowledge of rules of thumb to perform various jobs

4.3.4.8 Processes and procedural knowledge

The sequence of steps required to develop and maintain software is referred to as the software process. It sets out the technical and management framework for applying methods, tools and people to the software task (Humphrey, 1995). Process definition establishes measures and provides entry and exit criteria (Humphrey, 1995). By following various methodologies such as CMMI organisations rigorously define processes and adhere to them. Thus they increase their process and capability maturity levels. Formal software processes may not be defined in many SE organisations. In all these situations, the individuals will have an understanding of the processes to be followed to achieve outcomes. Literature review did not reveal operational items for this variable. Also, it is envisaged that the variable is simple enough to be understood directly by the respondents. Hence, the researcher operationalised processes and procedures directly into one operational item: The word software development was chosen instead of SE since, it is believed, that the word development was easier to understand amongst IT professionals. Further, the words software development has been used in literature to denote SE (Henninger, 1997; Sawyer and Guinan, 1998; Perry et al., 2001; Downey, 2005; Ye, 2006). The finalised operational item is as follows:

- Knowledge of software development processes used in the software project

4.3.4.9 Systems knowledge

Systems knowledge is defined as an understanding of the interplay of cause-and-effect relationships that is essential for sound decision making (Beazley et al., 2002). Todd et al. (1995) define systems knowledge as that category of knowledge that captures problem-solving skills including analytical and modelling skills as well as knowledge of development methodologies and systems analysis/design tools and techniques. Based on this definition, systems knowledge is operationalised into three parts:

- General problem solving skills
- Analytical and modelling skills
- Knowledge of development methodologies used to build the application system

4.3.5 Operationalisation of variables 2: Performance

WordNet (Princeton University, 2007) defines performance *"as the act of performing; of doing something successfully; using knowledge as distinguished from merely possessing it".*

Performance is defined as the manner in which or the efficiency with which something reacts or fulfils its intended purpose (Dictionary.com, 2007). In the case of SE, individuals possess knowledge, skills and abilities. They perform work to develop, enhance or maintain the application software system by using this knowledge. The manner in which or the efficiency with which they are able to complete the SE task is referred to as the performance. Operationalising performance is always a difficult issue for there are various types of performance. Performance is a measure of success (DeLone and McLean, 1992). Self-reported performances have been used in the literature to measure performance (White and Leifer, 1986; Rasch and Tosi, 1992; Wade and Parent, 2001) and can be used where there are no biases and self-gain. Further, they have been found to correlate with management reported performance (Rasch and Tosi, 1992).

The second research question only captures the contribution of different knowledge types to performance. This research question neither makes any effort to judge the actual performance of the individual in their job nor in any way is related to any personal gain for participants. Hence, it can be expected that individuals will not have any biases in answering this question. Hence, this research uses a self-reported perceived contribution of knowledge variables to improved performance. The following item is captured to answer the relation between knowledge and performance.

- The perceived contribution of knowledge types to improvement in performance

In order to keep the number of operational items on the final research instrument lower and to reduce complexity, the knowledge variables are directly operationalised along with brief descriptions of each of the variables. Such an approach has been adopted in literature before, by Nansi (1998), where Zmud's (1983) classifications were directly operationalised. Hence, the nine knowledge variables identified in the theoretical framework were directly operationalised to answer the research question relating to performance. In addition, the research instrument captures other information (increase in knowledge for the different knowledge variables and increase in performance).

4.3.5.1 Other operational items for case study

It is known that dispositional factors have influences on performance (Rasch and Tosi, 1992; Perkins et al., 1993; Billett, 1995). Hence, to obtain contextual perspectives from case study participants, the case study questionnaire incorporates additional dispositional items identified in Rasch and Tosi's (1992), as follows:

- Effort
- Achievement needs
- Self-esteem
- Personal reasons

4.3.6 Demographic profile

Extensive demographic profile information was collected in the research instrument to establish any relationships. These are summarised here.

4.3.6.1 Individual specific details

The overall experience of the individual is an important contextual factor relating to knowledge and performance (Chung and Guinan, 1994; Hunter, 1994; Nord and Nord, 1997; Sawyer and Guinan, 1998; Ng and Feldman, 2008; Dokko et al., 2009). During the work on identifying the skill needs, Sawyer et al. (1998) identified the importance of experience of working in the SE team. Hence, following variables are incorporated as follows:

- Overall number of years of experience
- Experience within the software project team

The literature review indicates differences in skills and knowledge levels and gender are related (Teo and Lim, 1996; Igbaria and Chidambaram, 1997; Gallivan, 2004). Hence, gender is added as a demographic profile item.

The following other demographic profile items were added:

- Mode of work such as Predominantly work at the organisation’s location with other members of the team, work remotely from home
- Highest qualification
- Field of study

4.3.6.2 Relative time spent on SDLC phases

As an addendum, the researcher has an interest in finding the knowledge requirements for various phases in SE and hence captured information on relative time spent on various SDLC phases. Adapting, the (Boehm, 1981) SDLC model and modifying it for this research, following phases were included in the questionnaire:

- Requirements analysis

- Architectural design
- High level design
- Low level design
- Coding and unit testing
- Function testing
- Integration testing
- Configuration management
- Quality assurance
- Preparation of manuals
- Implementation
- Production support
- Project Management

4.3.6.3 Organisational environment

The following demographic profile items were captured about the organisational environment.

- City
- Industry segment of the application software
- Adherence to Software process model or standard in the organisation
- Formal accreditation/certification levels of the organisation
- Opinion on maturity level of software development processes
- Does organisation have defined standards or guidelines?
- Are knowledge locator tools used in the organisation?
- Technologies used to develop the application system

4.3.6.4 Roles and responsibilities

It was important to capture the responsibilities of the role to understand the work carried out by the role. Hence, a free form question on duties and responsibilities was added to the list of demographic profile items.

4.3.7 Summary

The theoretical framework developed in Chapter 3 has been operationalised into various operational items. The research makes use of operational items from existing literature.

Where such items are not available or not suitable for this research, operational items are formulated based on related concepts and in consultation with experts in the field. Based on literature, the *soft skill* has been operationalised into items such as Verbal communication skills, Inter-team communication skills, Investigative skills, Organisational skills. The *technique skill* has been operationalised to items such as business analysis and software skills to develop the application. The *configuration knowledge* has been operationalised by making use of the related concept of function point analysis. The operational items for configuration knowledge include knowledge of user interfaces, knowledge of interfaces, database structures, program functionality and SE environment.

By adapting the communication framework developed by Curtis et al. (1988), the *social network knowledge* is split into three items: 1) social network knowledge within team, 2) outside the team but within the organisation, 3) social network knowledge outside the organisation. The *cultural knowledge* has been operationalised into eight items. These include 1) attention to detail, 2) Innovation and risk taking, 3) Outcome orientation, 4) People orientation, 5) Team orientation, and Aggressiveness, 7) stability and 8) Knowledge sharing culture. The *heuristics knowledge* was operationalised based on a study of literature into 1) knowledge of shortcuts, 2) standards, 3) guidelines and 4) rules of thumb. Similarly, *systems knowledge* was operationalised to three items: knowledge of development methodologies used to build the application system, problem solving skills and analytical modelling skills. Operationalisation of some of the variables in the case study research, such as configuration knowledge, cultural knowledge, social network knowledge and heuristics knowledge, will make use of the actual contextual knowledge in the organisational circumstance. The variables *process knowledge* and *business domain area knowledge* were operationalised directly into items.

In order to understand the contribution of knowledge type to improvements in performance, the following operationalisation was carried out.

- For every knowledge variable identified in the theoretical framework, perceptions of the individuals on the contribution of increased knowledge to improvements in performance are captured.

Further to the core data required to answer the research questions, the research collects number of demographic profile data such as city, Experience within SE team, Total Professional Experience, amount of time worked in different SDLC phases, roles and responsibilities. It is expected that by collecting these demographic profile information, the research would be able to draw findings on any relation with demographic profile variables.

4.4 General-survey research design

4.4.1 Introduction

Survey research gathers information about opinions and perceptions from a fraction of the population studied (Scheuren, 2007) to quantitatively arrive at findings. A written questionnaire based non-probabilistic survey is designed. The detailed questionnaire design for survey based on the operationalisation of variables is reported in section 4.4.2. The pilot study that was undertaken to revise the questionnaire and ensure its validity is described in 4.4.3. The general-survey intends to use non-probabilistic sampling to collect data. This sampling design is presented in section 4.4.4. The data collection methods and procedures for the general-survey are presented in section 4.4.5. The data analysis plan is specified in section 4.4.6.

4.4.2 Questionnaire design

Based on the theoretical framework and operational items that have been defined in section 4.3 a detailed questionnaire is prepared (see Appendix 11.3). The researcher took various samples of questionnaires (Nelson, 1991; Nansi and Bennett, 1998; Sawyer and Guinan, 1998) to understand the look and feel and used them during questionnaire development process. Part-A of the questionnaire captures the demographic profile items as identified in section 4.3.6. Part-B of the questionnaire implements the data collection strategies for the research question on knowledge required to perform various SE roles. For every operational knowledge item on the questionnaire, the research captures the extent to which that knowledge item was required to perform their job. A sample of Part-B of the questionnaire is shown in Figure 4.3 for illustrative purposes. Part-C implements the data collection strategies relating to the research question on knowledge variables that contribute to improvements in performance. To answer the research question, the following key operational items are captured in Part-C:

- For every operational knowledge type that has been identified in the theoretical framework, following information will be captured:
    - Contribution of that operational knowledge type to increase in performance.

The relative contribution to increase in your performance (column C) is captured using a four level Likert-type scale. In addition Part-C captures the following data items: 1) overall perceived increase in performance Question C.1 and 2) relative increase in knowledge (Column B).

Qualitative data is captured throughout the survey questionnaire including 1) questions on other types of operational knowledge item (see last question item in Part-B Appendix 11.3), 2) examples of scenarios involving each of the knowledge items and how they contributed to job performance (see question C.3 Appendix 11.3). Several considerations underpinned the questionnaire design. These are discussed as follows.

Open ended or fixed-alternative questions

Fixed-alternative questions, with a preset list of answers, were used for most questions as it was possible to structure alternate answers due to well formed theoretical framework and operationalisation (McGaw and Watson, 1976). Free format open ended questions that are not followed by any choice (Nachmias and Nachmias, 1976) were used to solicit qualitative information and close any incompleteness in fixed-alternative questions.

Structure of questionnaire

The questionnaire is separated into three parts with logically similar items grouped together to provide as much self-clarity to the respondent. The structure of the questionnaire is as follows (see general-survey questionnaire in Appendix 11.3):

- Part A of the questionnaire contains details relating to demographic and work experience profile of the individual.
- Part B of the questionnaire is related to the extent to which knowledge items helped in performing the job (see a sample part in Figure 4.3).
- Part C is related to reasons for increase in performance from a knowledge point of view

PART B: Knowledge items required to perform your job.

Please indicate the extent to which the following items help you to perform your main role successfully

Please circle the chosen option for each item.

	Not applicable	little help	moderate help	great help	necessary
Team work skills (long term)	•	•	•	•	•
Time management skills	•	•	•	•	•
Adaptability to new technologies & languages	•	•	•	•	•
Verbal communication skills	•	•	•	•	•
Inter-team communication	•	•	•	•	•
Investigative skills	•	•	•	•	•
Organisational skills	•	•	•	•	•
Ability to give and receive constructive criticism	•	•	•	•	•

Figure 4.3. Sample of Part B of questionnaire to collect knowledge requirements

Question wording

Appropriate guidelines on question wording, such as 1) no ambiguity, 2) no bias on alternatives, 3) social desirability, 4) no double-barrelled questions, 5) simple and short and 6) language and wording (McGaw and Watson, 1976; Sekaran, 2003) were adhered to during question design. For example, initial design of the questions contained double-barrel type questions – 'knowledge of standards and guidelines'. These were later revised to two separate questionnaire items to 1) knowledge of standards and 2) knowledge of guidelines.

Appropriate scales were used in the research questionnaire[29]. In the general-survey questionnaire, nominal scale was appropriate to classify and measure variables such as highest qualification (Question A.4), Role (Question A.6) (see questionnaire in Appendix 11.3). Interval scale was used to elicit responses on the differing magnitudes and proportions of knowledge needed to perform job (Part B and C). This research questionnaire uses different types of rating scales to elicit responses from respondents. The key ones and relevant examples from the general-survey questionnaire (see Appendix 11.3) are presented below:

- Dichotomous scale: This scale is used to elicit a Yes or No answer. In this general-survey questions such as 'Does your project team has defined standards and guidelines?' (Question: A.16) used the dichotomous scale.
- Likert scale – The Likert scale is designed to examine how strongly subjects agree or disagree with statements on a 5 point scale with different anchors. For example, the extent to which the knowledge item helped in performing the job is captured using a five-level Likert-type scale (Question in Part-B; see Figure 4.3).
- Itemized rating scale: The research elicits responses using itemized rating scale for questions in Part C on knowledge and performance (see question C2).

Scale points

For each of the variables which use the interval scale the number of scale points needed to be decided. Davenport (1997) presenting on Microsoft's skills and competency management system, has reported using a four point scale on expertise with scale points as 1) basic, 2)

---

[29] Sekaran (2003) classifies scales into four basic types. Nominal, interval, ordinal and ratio scales. The nominal scale allows the researcher to assign subjects to certain categories or groups while the ordinal scale rank orders the categories of the nominal scale. The interval scale provides more information on magnitude of the differences and their proportion. Ratio scales provide for much more precision.

working, 3) leadership and 4) expert. Shi Nansi (1998) analysing the knowledge requirements for systems managers has used a five point scale. While the general-survey is conducted in various countries, all the respondents are professionals working in enterprise SE roles. Hence, a five point scale for identifying knowledge needs for performing various roles with the scale points 1) not applicable, 2) little help 3) moderate help, 4) great help and 5) necessary is used (see Figure 4.3). For the purpose of identifying the increase in knowledge and contribution to increase in performance (Question C2-B in Part-C) a four point scale was preferred with the scale points 1) nil, 2) small, 3) moderate and 4) great. During pilot testing the scales were reviewed for feedback with the participants. There were no significant comments that warranted a change to the scale points.

4.4.3 Pilot study of general-survey questionnaire

A pilot study can be the pre-testing or trying out of a particular research instrument (Baker, 1994, cited in Teijlingen and Hundley, 2001). The term pilot study can also refer to so-called feasibility studies or trial runs, done in preparation for the major study (Teijlingen and Hundley, 2001). One of the advantages of conducting a pilot study is that it might give advance warning about where the main research project could fail, where research protocols may not be followed, or whether proposed methods or instruments are inappropriate or too complicated (Teijlingen and Hundley, 2001). The words of De Vaus (1993 Teijlingen and Hundley, 2001) have provided caution to this researcher '*Do not take the risk. Pilot test first.*' Pilot studies provide us an insight into the research process and analysis. They provide the researcher with the view point of the participants. It provides an opportunity to get the research instrument right. Also, it provides an opportunity for the researcher to analyse the resulting data and understand the direction. The terminologies pre-test and pilot study, have been used interchangeably in the literature. Pilot studies can include tests to ensure validity and reliability. Validity is concerned with the extent to which the research data collected is a true picture of what is being studied (Collis, 2003). On the other hand reliability is concerned with the repeatability of research findings.

Face validity addresses the concern of whether the questionnaire appears to measure the concepts being investigated (Cavana, 2001). In particular, the researcher is interested in finding whether the wording of the items is clear and is understandable to the participants. In order to achieve face validity, a small group of respondents are asked to answer the questionnaire. During answering of the questionnaire, the respondents are asked to provide feedback on the items for clarity, confusion, concepts and understanding. At the end of answering the questionnaire, the respondents are asked to summarise their experience. The questions are corrected or re-worded and the process is repeated until both respondents and the researcher are reasonably happy with the questionnaire (Cavana, 2001). Content validity relates to the representativeness of sampling adequacy of the questionnaire regarding the content or the theoretical constructs to be measured (Cavana, 2001). In order to achieve content validity, expert review is undertaken. Researcher reports the origins of each of the operational item, based on literature search. A group of experts examine each item and make a judgement on whether each item does measure the theoretical construct nominated (Cavana, 2001). No effort was made to test repeatability of test results.

The general-survey questionnaire prepared was piloted with a small sample of chosen SE professionals. The following objectives are set forth for the pilot study with two phases.

- Pre-test-1
  - Ensure the items in the questionnaire are face valid.
  - To make sure the data from the study can be analysed to a conclusion.
- Pre-test-2
  - Ensure the questionnaire follow the rules of content validity.

Team members from University of Canberra - SE teams were approached for this pre-test. Willing members were provided with the questionnaire and were encouraged to speak up their mind during answering of questionnaires on any issue including clarity, language, structure and reason. Extensive revisions were made to the questionnaire based on these comments and feedback from pilot study participants. The pilot study data was entered into Microsoft Excel and SPSS and the results were analysed. The researcher was satisfied about the possibility of analysing the resultant general-survey data.

Pre-test 2: Content validity tests

A panel of experts in the area of SE/KM are identified. This panel of experts was asked to assess the questionnaire with respect to items and their relation to the theoretical constructs they represent. The experts analysed every item for its adequacy in representing the theoretical construct that it represents. The experts were chosen from IT industry and academia with substantial SE and/or KM background. The research questions, general-survey questionnaires and the associated theory were passed onto the two members of the panel of experts. Experts were asked to review questions items and provide judgement on the following counts:

- *Whether the items fit the theories that they are supposed to measure.*
- *Whether the items are adequate and represent the theoretical concepts they are supposed to measure."*

Clarifications were provided to experts. Both experts confirmed that the items fit the theories they are supposed to measure. Also, they confirmed that the items were reasonably adequate and represent the theoretical concepts they were supposed to measure. One of the expert opined that there was always scope to add more items onto the questionnaire, however the items present on the questionnaire is exhaustive and is sufficient for the purpose of the research.

4.4.4 General-survey sampling design

The set of all members of enterprise application SE teams performing various roles would constitute the population of the general-survey for this project. Sample is a subset of the population that is used to gain information about the entire population (Henry, 1990). Probabilistic and non-probabilistic sampling are two ways to sample the entire population. In case of probability sampling, it should be possible to define the full population and be able to randomly sample that population, thus minimising bias and error. In order for the research to use probability sampling a list of all population members should be available. There is no list or database of all SE professionals available to the researcher. Further, existing lists[30] are not representative of the target population (SE professionals for this research) and are not accessible due to privacy reasons. The results from studies using non-probability sampling cannot be fully generalised to the population. Even adopting a probabilistic study is less likely to have increased the extent of generalisability due to issues such as low response rates, non-availability or refusal of participants (2007). Hence, general-survey uses non-probabilistic sampling.

Judgement sampling or purposive sampling involves the choice of subjects who are most advantageously placed or in the best position to provide the information required (Sekaran, 2003). This research is to aims to collect perceptions on knowledge required to perform the

[30] examples: members database of Australian Computer Society (ACS) or Microsoft Certified Professionals)

SE roles and contributions of knowledge to improved performance. Opinions from sample software professionals, even if not random, would be appropriate due to their specialist nature (Neuman, 2006) and is of immense help in understanding the knowledge requirements for SE roles. Hence, judgemental sampling technique is used in this research due to issues such as practical difficulties in defining and accessing a master list of population due to privacy issues.

There are number of limitations of non-probability sampling. These should be acknowledged and the risks of such sampling to generalisability of results are understood by the researcher and reported in the thesis. Sekaran (2003) says non-probability sampling is not generalisable. The subjective nature of selection of samples in case of non-probability sampling is a serious weakness for the study (Henry, 1990). Hence, there is a risk that the findings may not be valid because of bias in the selection process. However, this research carries out a well documented, systematic, empirical study based on a well-founded theoretical framework. The research aims to obtain a reasonable amount of judgemental samples that is acceptable to the SE discipline so that meaningful statistical analysis can be conducted. Hence, the results of the study have a degree of generalisability due to the careful collection of representative samples.

4.4.5 Data Collection Method

The researcher is a consultant working in the SE industry for more than fifteen years and has developed numerous professional and personal relationships of SE professionals around the world. These relationships were leveraged to collect the data. About forty contacts were sent an email to ask for help regarding collecting data from their application SE teams. The researcher also telephoned these forty members to ask for help. Volunteers from this group were designated as Data Collection Contact (DCC) subject to the following eligibility criteria.

- The DCCs must work in an enterprise application SE team.
- They should be willing to collect the data for this research.
- They should be ready to make use of their personal and professional contacts to collect responses from survey questionnaire.

The DCCs were provided briefing on data collection procedures, using telephone and emails, including on issues such as eligibility of participants, logistics and ethics.

4.4.6 Analysis plan for general-survey results

There are three parts in the general-survey as follows:

- Part A: Demographic profile including experience
- Part B: Knowledge items required to perform your job
- Part C: Why performance increases over time?

This section describes the analysis plan for general-survey results using each of these sections so that the research questions can be answered.

Quantitative summary of demographic profile information will be presented. Part: B of the general-survey collects knowledge items required to perform various roles in the SE team. The results will be collectively summarised using mean and standard deviation for each variable in the theoretical framework. Reliability (Cronbach's alpha) measure will be reported for the theoretical framework variables. The qualitative data will be summarised and presented. Thus, the summary quantitative statistics and qualitative summary will provide evidence to justify that the different knowledge types identified in the theoretical framework contribute to job performance.

For every knowledge variable identified in the theoretical framework (nine types of knowledge), perceptions of the individuals on the contribution of increased knowledge to improved performance is captured in Part-C (Question C.2.B.n; where n denotes the identifier for knowledge type). Rankings will be assigned based on the mean of *perceived contribution of the increase in knowledge to improved performance*. Based on this analysis, it is expected that the researcher will be able to answer the relative importance of different types of knowledge to improvements in performance. Qualitative evidence gathered in the general-survey will be summarised to provide substantiation to the quantitative findings. Note that additional information captured in Part C (Overall increase in performance and increase in knowledge) of the questionnaire is not used for analysis purposes.

4.4.6.1 Re-classification

Exploratory Factor Analysis (EFA) is a heuristic method for classification or identifying causal factors behind a set of data. Common components can be identified that identify patterns of underlying relationships or causal connections in data (Rummel, 1970). Factor analysis is a tool that has been widely used in SE/IS research (examples: (Nelson, 1991;

Campbell, 1997; Rajeswari and Anantharaman, 2003; Shi, 2007)). EFA will be employed as a tool against the data collected on knowledge requirements to understand the underlying pattern of factors. This analysis has the capability to arrive at a re-classified theoretical framework. The reliability (Cronbach's alpha) for various factors identified through factor analysis will be reported.

Analysis of Variance (ANOVA) will be carried out between demographic profiles and each of the knowledge factors that have been identified. This will provide insight into any difference in perceptions between participants of different environmental context and background. Multiple regression analysis will be carried out between these different types of experience (Total Professional Experience and Experience within SE team) and the extent to which various knowledge factors were needed to perform the job showing any relationships between experience and knowledge requirements. Only significant statistical relationships posing logical sense to the underlying theory will be reported in the thesis.

4.4.7 Limitations of the study

Limitations of the general-survey research method are presented in this sub-section. The sampling methodology used in this study is non-probabilistic judgemental sampling approach, due to which bias could be introduced. The sample has been collected by using researcher's contacts and relationships built over time. In spite of this limitation, this research is useful to the wider industry and academia. Number of other research has been carried out using non-probabilistic sampling. Other studies in SE/IS field have used non-probabilistic sampling approach to address similar problems (examples : (Kraemer and Dutton, 1991; Nansi and Bennett, 1998)). Hence, it is argued that using non-probabilistic sampling of the nature described in this thesis is acceptable to the discipline.

Different positions with same role title, within different organisations (even within the same organisation) involve different requirements for knowledge and may involve substantial variation in job specifications and hence the knowledge requirements. Any generalised theoretical framework that is developed by using this general-survey won't apply to a given position in a specific organisation directly. A knowledge needs analysis should be performed for a given position based on the results of this research to understand the knowledge requirements for that position.

The study collects opinions on knowledge requirements for performing various roles and performance improvements from the participants. Based on the data collected the study arrives at the knowledge requirements to performing the various roles. The perceived knowledge requirements from individuals could be different from the real knowledge requirements for the job. Also, there may have been an ideal (Hunter and Palvia, 1996) set of knowledge requirements to perform these roles, which is not known or not practised by the professionals in the industry. Further, tacit knowledge that is inarticulable (Busch et al., 2001) cannot be collected in a survey.

The general-survey collects opinions at a certain point in time and reports the results. The knowledge requirements and performance increases can change in the future due to various reasons (example: technological change, changes to role definitions). Any variations in future must be researched and reported at that time.

4.4.8 Summary

The research aim is to reveal the knowledge requirements and variables contributing to improved performance for SE roles. This research proposes to conduct an analytic, written based, perception survey to answer the research questions. Questionnaire was designed using items identified in the operationalisation of variables (see section 4.3). The questionnaire design made use of the best practices suggested from literature such as the ones on wording, types of questions and structure of the questionnaire. Most of the questions were fixed alternative questions. However, some of them were open-ended questions to obtain qualitative feedback from participants. Appropriate scales were assigned. Number of demographic profile items, were captured in the questionnaire. The developed questionnaire was pilot tested with practitioners. Revisions to questionnaire were made based on comments and feedback from participants. A content validity study was conducted wherein academics verified that the items in the questionnaire indeed represented the theoretical constructs that they are supposed to measure.

Due to difficulties in obtaining a master database of population, such as privacy and accessibility of data, the research uses a non-probabilistic judgemental sampling methodology. Non-probability sampling poses risks to generalisation of the results. Nonetheless, the results provide substantial value since it is carried out amongst a carefully targeted and expert sample. The data collection plan specifies that the contacts established by the researcher who work in SE teams around the world are used to collect data. These volunteers, titled Data Collection Contacts (DCC) will collect the data from the fellow members of their SE teams. Data collected from the general-survey will be analysed using both descriptive statistics (such as mean, standard deviation) and inferential statistics (such as ANOVA, correlation). The qualitative data will be summarised to provide additional evidence in answering the research questions.

Number of limitations, exist for this research study. Firstly, the general-survey uses a non-probabilistic sampling methodology, which poses substantial risks to generalisation. The general-survey only collects perceptions of knowledge and performance. Such perceptions may not be real. The knowledge requirements for the same role type in different organisations may vary. The general-survey collects information in a snapshot of time. There can be variations to the knowledge requirements and variables in the future and will need to be investigated by separate research.

4.5 Case study research protocol

4.5.1 Introduction

Case studies are a form of naturalistic and empirical inquiry to study complex phenomenon in the real environment (Middleton, 1985, cited in Nansi, 1998; Yin, 2003). Case studies are appropriate for sticky practice-based problems where experience of actors is important Bonoma (1983, cited in Benbasat et al., 1987). By its very nature, case studies are natural and attempt to depict reality and allow triangulation of data from multiple sources. Disadvantages include issues relating to 1) the researcher's bias, knowledge, skills and experience and 2) the ability to generalise the results.

The overall case study research design covering the detailed rationale behind choosing case study research method, the nature of the case study research, the bias of the researcher and criteria for selection of field site are presented in section 4.5.2. The data collection plan including 1) data items gathered, 2) the project-survey questionnaire design, 3) sources and methods of data collection are presented in section 4.5.3. The data analysis plan is specified in section 4.5.4. Ethical issues for the case study research, in an organisational context warrant a higher focus than an anonymous survey and hence, are presented in a separate section 4.5.5. In this case study research, the researcher also played the role of a key informant in providing key data. The issue of bias created due to this arrangement and key advantages and disadvantages of such an approach are discussed in section 4.5.6.

4.5.2 Overall case study research design

4.5.2.1 Rationale for the case study research method

The research carries out a single-embedded case research of a typical SE team. This case study research is aimed at depicting a real life situation. Lessons learned from the case study are informative about the experiences of the average person or institution (Yin, 2003). Further, the case study research described in this thesis is founded on a theoretical framework (see Chapter 3). The formulated theoretical framework is tested in typical SE team. The findings from the case study can be informative, and can shed light into a real life context and provide empirical evidence (Yin, 2003) to the proposed theoretical framework in this research

Yin (2003) proposes four types of case study designs: 1) holistic single-case, 2) single-case with embedded units of analysis, 3) holistic multiple-case and 4) Multiple-case with each involving embedded units of analysis. A chosen SE team forms part of a single-case. The software project team consists of many individuals performing different roles. The research question attempts to find the knowledge requirements and variables contributing to performance of an individual performing the role in a SE team. The unit of analysis is at an individual level. Hence, this research carries out a single-case with embedded units of analysis.

Burns (2000) suggests three criteria for selecting cases: typical, ideal and available. Based on this, following criteria are developed for selection of case study SE team.

- Part of a large commercial or government organisation
- Involved in enterprise application software development, maintenance or support
- Team size of more than 10. This team size is stipulated to get a better quantity of data that is suitable for analysis.
- Accessible for conducting the research

At the time of developing the proposal, the researcher was working as a consultant-developer/lead-designer in an application software project team within a large government organisation. The researcher approached the management of the organisation to see if access to the software project team could be granted for the purpose of this research. The team fitted all the criteria required for aspects of a typical SE team. The organisation granted the required access (see section 4.5.5 for ethical considerations and conditions of access). The individuals in the team agreed to participate in the research (see informed consent form in Appendix 11.2) and hence the site was chosen as the field site for the case study research.

4.5.3 Data collection

Yin (2003) identifies six sources of data that are predominantly used in case study research. These are documentation, archival records, interviews, direct-observations, participant-observation and physical artefacts. Interviews are an essential source of evidence since most studies are about human affairs and human activity (Burns, 2000; Yin, 2003). The nature of this research is about knowledge and action. Hence, interviews are chosen for this case study research. The case study research uses three methods of data collection as follows:

- Project profile from key informant: A key informant provides information required for project profile. This project profile information is used to inform the operational items developed earlier in the thesis.

- Project-survey questionnaire: Case study research questionnaire is provided to the member of the software project team. The participant is asked to fill up the project-survey questionnaire.
- Face-to-face Interview with the respondents.

Project profile from key informant

The case study research makes use of key informant technique. Yin (2003) says that when a respondent can also suggest other persons for interview, as well as other sources of evidence, the more that the role may be considered of an informant rather than a respondent. With the help of the key informant, the following types of contextual knowledge relevant to the project team are collected.

- Technical skills that are generally required for performing work on the system
- A list of application software items in the project team.
- Social networks formed with various teams around the project team within the organisation.
- Any relevant explicit cultural knowledge items present in the organisation.

In this case study research, the researcher acted as the key-informant as well. The researcher having worked in the case study software development project team for more than three years knew the contextual knowledge items including application system and people. The researcher consulted with expert colleagues before finalising the project profile information, as a mechanism to corroborate the information. The project profile information is used to inform the project-survey questionnaire.

Project-survey questionnaire data collection

Survey-type interview is a type of interview with structured questions that is along the lines of a survey. Yin (2003) says such a survey could be designed as part of a case study and produce quantitative data as part of the case study evidence. This case study research makes use of a structured interview, titled project-survey (see appendix section 11.3 for the template project-survey questionnaire). The project-survey questionnaire was administered to participants who provided informed consent (see ethical aspects of research in section 4.5.5) to participation.

This case study research project aims to answer two research questions.

- Part I : Operational knowledge types required to perform various roles in software development projects

- Part II: Identification of knowledge factors that contribute to increase in performance.

In part I, for every operational knowledge item, on the project-survey questionnaire the extent to which that knowledge item was required to perform their job is captured. During this process, two additional data elements are also collected: 1) Initial knowledge and 2) Current knowledge. Initial knowledge and Current knowledge are measures of the individual's knowledge at the time the individual joined the project team and as at the time of writing the project-survey questionnaire respectively. These two additional data elements will be used for analysis of Part II.

Part-II of the questionnaire is aimed at identifying the knowledge variables that contribute to increase in performance. Based on the theoretical framework, it is envisaged that any increase in a knowledge type should have corresponding contributions to increase in performance. Hence, the questionnaire captures following data elements.

- The overall increase in performance is captured in the survey.
- For every operational type that has been identified in the theoretical framework, following information is captured:
    - Contribution of that operational knowledge type to increase in performance.

Extensive qualitative data was captured in the data collection phase including 1) examples of various types of operational knowledge items, 2) scenarios involving each of the knowledge items and how they contributed to job performance. The design of project-survey questionnaire in this section.

The design of the general-survey questionnaire was discussed in 4.4.2. This general-survey questionnaire has been prepared based on the theoretical framework and operational items that have been developed in section 4.3. The general-survey questionnaire is informed by the context-specific knowledge items that have been identified in the project profile phase. The social network knowledge part of the project-survey questionnaire lists all the identified teams with which the project team interacts within the organisation. The configuration knowledge part of the question is informed by the list of software items identified in project profile phase. A sample of Part-I of the project-survey questionnaire is shown for illustrative purposes in Table 4.1. For each knowledge item, the project-survey questionnaire asks for the following three different data elements:

- Knowledge/Skill rating at the time the individual joined the project. (C1)
- Current rating of knowledge/skill (C2)
- Extent to which this knowledge helped in performing the job (C3)

Full project-survey questionnaire is shown in Appendix 11.3.

The overall perceived increase in performance is captured by one item on the project-survey questionnaire in question B.11 (see section 11.3 for project-survey questionnaire template). The contribution of different variables contributing to increase in performance is captured using question B12. These are derived from the theoretical framework that has been developed in this thesis. Apart from the variables identified in the theoretical framework, further dispositional factors identified by Rasch and Tosi (1992) added to the list. The final list of knowledge variables identified in the case study project-survey questionnaire is as follows:

- Increase in your technique skills
- Increase in your soft skills
- Increase in your business domain area knowledge
- Increase in your knowledge of processes and procedures
- Achievement needs
- Increase in your systems knowledge
- Increase in your social Network knowledge
- Increase in your cultural Knowledge
- Increase in your heuristics knowledge
- Increase in your knowledge of Application Software Items
- Self-esteem
- Goal clarity
- Goal difficulty
- Personal reasons

Part B : Knowledge

Identify and rate the knowledge required to perform your job in the application software development project team.

Column : C1 : Rate your knowledge/skill at the time you joined the project, first time.

Please use the following rating scale.

0 = No knowledge, 1 – Basic, 2 – Competent, 3 – Expert

Column : C2: Rate your knowledge/skill as of now. Use the above rating scale.

Column : C3 – To what extent this knowledge helped you in your work on the work package. Use the following rating scale

0 = nil, 1 = small, 2 = moderate, 3 = great

B.1 Technique skills

Skills that are directly required to performing the particular job. Examples : A programmer generally requires technical skills on languages, compilers, operating systems. Business analyst requires business analysis skills. Project manager needs project management skills.

Please circle the chosen option.

Knowledge/Skill	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 – Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 – Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
Project management skills	0 1 2 3	0 1 2 3	0 1 2 3
Knowledge of development methodologies used to build the application system examples : Waterfall model and spiral model	0 1 2 3	0 1 2 3	0 1 2 3
Knowledge of application radial architecture	0 1 2 3	0 1 2 3	0 1 2 3

Table 4.1. Sample part of case study project-survey questionnaire

Throughout the case study project-survey questionnaire qualitative data is collected (example: question B10 captures any other knowledge required to perform your job). For each of the questions, the participant is encouraged to write up thoughts on different types of knowledge, how it is required to perform the role, how it helped to increase in performance etc.

Interviews

The interview method of data collection was used as a mechanism to run through the answers provided by the participants and give them an opportunity to think through the answers and give comments. The interviews were designed to be guided conversations with respondents rather than structured interviews. The objective was to follow the filled case study research project-survey questionnaire by flowing through the list of questions presented in the survey method and seek additional comments, if any are available. The interview is intended to ensure that all answers are captured in the questionnaire and additional qualitative information is captured. Additional un-structured qualitative information, including opinions and examples of situations, are collected using this interview process.

4.5.4 Analysis plan

This case study is guided by a well-formulated theoretical framework (see Chapter 3). As suggested by Yin (2003) propositions from theoretical framework have guided the data collection plan and guide the analysis as well.

4.5.4.1 Knowledge requirements to perform various SE roles

Recall from the section on key data collected (see section 4.5.3) that for every operational knowledge item on the project-survey questionnaire the research captured the extent to which that knowledge item was required to perform their job (column C3 on project-survey questionnaire). This column C3 in the project-survey questionnaire helps us to answer the question on knowledge required to perform the job. The average extent to which the knowledge variables helped in performing their job is a measure of the extent to which that knowledge item contributed to performing their job for that role. A comparative analysis of knowledge types required to perform various roles will be made.

For configuration knowledge, a different analysis plan is adopted. Configuration knowledge is the knowledge of application system. Individuals performing various roles would require knowledge of software items at various levels. For example, a programmer may need to know in depth knowledge of some software items and some at a moderate level. From the data collected, average number of software items that were required to perform their job, at various knowledge requirements levels, for different roles, is summarised. The average number of items with different knowledge requirements levels is calculated for that role. These averages are normalised, to a percentage, by dividing with the total number of software items. The measure percentages of knowledge items needed for various knowledge requirements levels is an indication of the need for configuration knowledge to perform different roles. Further, quantitative data collected is summarised to provide additional support for the findings.

4.5.4.2 Knowledge variables contribution to improved performance

In the project-survey questionnaire, for each operational knowledge variable identified in the theoretical framework, following information is captured: (see section 4.4.1):

- Initial knowledge when the individual joined the project
- Current knowledge.
- Contribution of that operational knowledge type to increase in performance.

Increase in knowledge since joining the team can be computed by subtracting current knowledge and initial knowledge. This research proposes a simple knowledge index to quantify initial and current knowledge of the individual. The definition, appropriateness and reasoning behind usage of knowledge index for this research is presented in Appendix 11.1.

In summary, the following variables have been available for analysis of Part II. This information is quantitatively analysed to provide relevant findings.

- Variables contributing to performance (also referred to as reasons for increase in performance)
- Perceived contribution to increase in performance due to various knowledge variables
- The initial knowledge for the individual at the time the individual joined the project for all knowledge variables.[*]
- The current knowledge for the individual at the time the individual joined the project for all knowledge variables[*]
- Increase in knowledge for the individual for this type of knowledge [@]
- Individual name[#]
- Increase in performance
- Role of the individual

* - For the purpose of this quantitative study, knowledge of an individual is calculated using an index referred to as the knowledge index (see 11.1 for more details)

@ -Increase in knowledge is calculated by subtracting initial knowledge index from current knowledge index.

- Individual name will not be the actual name (see section on ethics 4.5.5 on page 102.)

Further, qualitative data collected is summarised to provide additional support for the findings.

4.5.5 Ethics

A submission on the case study research project was made to the ethics committee of the University of Canberra regarding the project and approval obtained. The case study is at a Government agency in Canberra. The case study uses interviews and analysis of existing documentation to collect data. Individuals will not be identified. The case study is proposed to begin in Jan-06 and run for six months. Quantitative and qualitative analysis is carried out on the resulting data. The researcher worked as a consultant in the same case study software project team at the Commonwealth agency.

Details about the participants

Individuals working in application SE projects are considered for this case study research. Expected age range is between 20 and 70. The project does not take into account the individual's personal cultural or social backgrounds of participants.

Recruitment of the participants

A software project team was approached to participate in the case study. All members of the chosen software project team would be considered as prospective participants and they will be approached for participation. The participants were not dependent on the researcher. No inducement was provided to the participants. Participants were asked to sign to a letter of informed consent before participation. A copy of the letter of informed consent is attached in section 11.2.

Privacy of data collected

The case study data is confidential and will be held in a shelf under lock and key and will be maintained by the researcher. This data will not be revealed to anyone else and will be destroyed at the end of the research. An optional e-mail address will be collected from the participants. At the end of the research an e-mail will be sent to all participants

Agreement of the organisation

The organisation agreed to let the researcher conduct the research on its premises and with its staff subject to the following conditions:

- No identifying information about the organisation or individuals should be released.
- Only summary information can be released for publication or thesis.
- The report must be vetted by an officer to make sure nothing references the organisation.

- Participation in the interview is voluntary.

4.5.6 Bias due to researcher as a participant and key informant

The researcher worked as a part of the case study team. This leads to a number of advantages and disadvantages. The advantages include access to inside information, established relationships with other participants and prior knowledge of information. On the other hand, participant bias can creep into the researcher's mind. Every effort is made to reduce this bias. These include the following:

- Empirical research with processes clearly defined
- A clear case study protocol is outlined before the start of research. This protocol was reviewed by the supervisor and approved.
- Ethical research committee approval
- The project-survey questionnaire prepared and pre-tested before being used.
- All the data collected are noted down on paper before being summarised
- Positivistic quantitative analysis of data rather than qualitative opinion based analysis of data. Qualitative data gathered is presented as additional evidence.

4.5.7 Summary

Case study research is helpful in investigating practice based problems. This research is concerned about the knowledge and performance of individuals in SE teams. Case study research will be able to provide an account of the real life context in answering the research questions. A single embedded, positivistic, predominantly quantitative case study is intended to be conducted and results presented. A SE team in a large organisation with more than 10 team members with accessibility was selected for carrying out the case study. The researcher was able to obtain the necessary permissions from the place where he worked.

An initial project profile of the SE team is gathered by using a key informant (the researcher). This project profile captured contextual knowledge. For example, the configuration knowledge associated with the SE team such as user interfaces, interfaces, functions were identified. This contextual information was then used to inform the general-survey questionnaire, to develop the project-survey questionnaire. An interview was organised wherein additional qualitative data was captured. Case study analysis makes use of descriptive statistics of the collected quantitative data to answer the research questions. The qualitative evidence is summarised to provide additional evidence in answering the research questions.

One of the key aspects of carrying out case study research in an organisational context is the ethics. Volunteers who participated in the case study signed a letter of informed consent. Privacy and anonymity of data was guaranteed not only to the individuals but to the organisation. The researcher worked in the SE team as a consultant. He also took the role of key informant for this research. In order to reduce participant bias, steps such as clearly defined and documented case study protocol before the study, positivistic predominantly quantitative approach and ethics committee approval were taken.

4.6 Chapter Summary

The research methods for answering the research questions are generally selected based on an analysis of the context, suitability of research methods and author's philosophical bent and capability. Being a practice based problem, the case study method was selected to provide a real life context in answering the research question. A survey is planned so that empirical quantitative evidence can be obtained to answer the research questions, allowing us to provide evidence to theoretical framework. Reconciliation between results of case study and survey is carried out.

The theoretical framework developed in Chapter 3 has been operationalised into various items. Based on literature, the soft skill has been operationalised into items such as verbal communication skills, inter-team communication skills, investigative skills and organisational skills. The technique skills have been operationalised based on the responsibilities of the role (examples 1) a programmer requires software skills to develop the application and 2) business analyst requires business analysis skills). The configuration knowledge has been operationalised by adapting and using a related concept of function point analysis. The operational items for configuration knowledge include knowledge of user interfaces, knowledge of interfaces, database structures, program functionality and SE environment. By adapting the communication framework developed by Curtis et al. (1988), the social network knowledge is split into three items: 1) social network knowledge within team, 2) outside the team but within the organisation, 3) outside the organisation. The cultural knowledge has been operationalised into seven items. Seven of these items were borrowed from organisational behaviour literature. These include 1) attention to detail, 2) innovation and risk taking, 3) Outcome orientation, 4) People orientation, 5) Team orientation, 6) Aggressiveness and 7) stability: Another item 8) knowledge sharing culture in the organisation is added to the cultural knowledge variable. The heuristics knowledge is operationalised based on a review of literature, into knowledge of shortcuts, standards, guidelines and rules of thumb. Similarly, systems knowledge was operationalised to items such as problem solving skills and analytical modelling skills. Operationalisation of some of the variables in the case study research, such as configuration knowledge, cultural knowledge and, social network knowledge will make use of the actual contextual knowledge in the organisational circumstance. For every knowledge variable identified in the theoretical framework, perceptions of the individuals on the contribution of increased knowledge to improved performance are captured. The research also collects number of demographic profile data such as city, Experience within SE team, Total Professional Experience, amount of time worked in different SDLC phases, roles and responsibilities.

General-survey Questionnaire is designed using the items identified in the operationalisation of variables phase. The questionnaire design made use of the best practices suggested from literature including on wording, types of questions and structure of the questionnaire. Most of the questions were fixed alternative questions. Pilot testing of questionnaire was carried out to ensure that data are able to be analysed and provide validity. A non-probabilistic sampling methodology is adapted to collect data. The use of non-probabilistic sampling poses risks to generalisation of the results from the study. The data collection plan specifies that the contacts established by the researcher who work in SE teams around the world would be used. It is intended that the data collected from the survey can be analysed using both descriptive statistics (such as mean, standard deviation) and inferential statistics (such as ANOVA, correlation). The qualitative data will be summarised to provide additional evidence in answering the research questions.

A single embedded, descriptive case study is intended to be conducted and results presented. A SE team in a large organisation with more than 10 team members, that was accessible, was selected for carrying out the case study research. The data collection methods include key-informant technique, questionnaire and interviews. Volunteers who participate in the case study are required to sign a letter of informed consent. Descriptive statistics is used to summarise quantitative data. The qualitative data is summarised to provide additional evidence in answering the research questions.

Both survey research and case study research make use of the same theoretical framework. Reconciliation between the two findings will be made. Further exploration of data collected from survey will be carried out using factor analysis to see any hidden factors in the data. The results will be interpreted in light of the underlying theories formulated in this thesis.

5 General-survey research findings

5.1 Introduction

This chapter reports the results of general-survey. This research set out to understand the knowledge requirements to perform various roles and the knowledge variables that contribute to improved performance. This chapter reports the results of the general-survey research (see section 4.4 for design) to gather evidence. Data was collected from samples of the specified population using a purposive-judgemental sampling methodology.

The organisation of this chapter is presented in Figure 5.1. The demographic profile of data is summarised in section 5.2. Section 5.3 presents the analysis of data on knowledge requirements to perform various SE roles. Section 5.4 presents the analysis of data on knowledge variables contributing to improved performance. Section 5.5 presents the analysis of the relation between demographic variables and the Contribution of knowledge to performance. The limitations are discussed in section 5.6. Finally, section 5.7 summarises the findings from the general-survey.

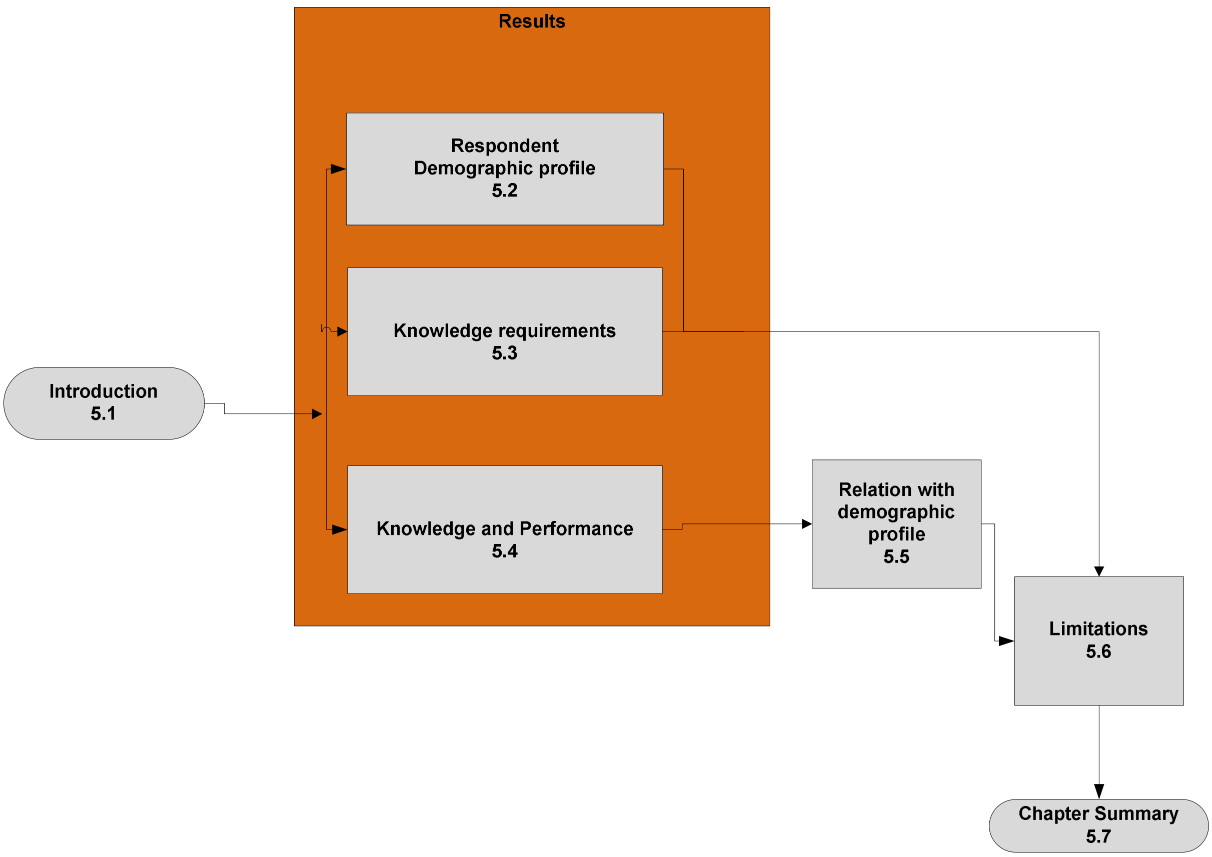

Figure 5.1. Flowchart: Organisation of Chapter 6

5.2 Analysis of Part A: Demographic profile

5.2.1 Introduction

The general-survey questionnaire (Part-A; see Appendix 11.4 for full questionnaire) collected demographic information such as individual characteristics (including gender, experience and education level) and organisational characteristics (such as industry segment and technologies). The questionnaire also collected responsibilities for performing SE roles. This section summarises these demographic profile information.

5.2.2 Respondent profile

Table 5.1 shows the role profile of respondents. The number of responses from programmer-analysts (78) tops the list followed by Programmers (28) and business analysts (28). It was harder to identify and collect responses from project managers, designers, testers and architects since there are few of them available to collect the responses. Nine respondents (5.05% of total) carried out combination roles and Eleven (6.1%) carried out other roles (see Table 11.2 and Table 11.3 in Appendix 11.13 for details). The number of male respondents (71.35%) in this general-survey is higher than the female (28.65%) respondents[31]. Table 5.2 shows the distribution of the number of respondents and their experience range. The number of participants with greater than 5 years of experience (121) outweighs the count with less than 5 years (49).

[31] This is nearly consistent with Australian Bureau of Statistics (ABS) survey (ABS, 2004), which shows 34% female participation in ICT workforce in Australia.

Role	Number of respondents	Percentage
1-BusinessAnalyst	26	14.60
2-Programmer	28	15.73
3-Programmer/ Analyst	78	43.82
4-Architect	3	1.69
5-Designer	5	2.81
6-Tester	5	2.81
7-ProjectManager	13	7.30
A1-Other roles	11	6.18
A2-Combination roles	9	5.06
Total	178	100

Table 5.1. Role profile of respondents

Experience band	Number of respondents
Less than or equal to 1 year	10
Greater than one year and less or equal to two years	7
Greater than two years and less or equal to three years	13
Greater than three years and less or equal to five years	19
Greater than five years and less or equal to ten years	59
Greater than ten years	62
Missing data	8
Total	178

Table 5.2. Experience profile

Majority (113; 63.4%) of the responses have been collected from Canberra. Twenty five of the responses were collected from Chennai. The section 11.15 (Table 11.4) shows that 20 responses were collected from USA and 4 from Singapore. Amongst the respondents 50.56% of them had a Bachelor degree with a further 32% possessing Master's degree.

5.2.3 Respondent's Organisational profile

More than half of respondents came from a Government/public sector background (57.8%) while others worked in private sector domains such as banking, telecom and media. CMMI has been adopted in many (34.83%) of the organisations of the participants. ISO follows next with 15.17%. However, very few of them have attained any formal accreditation levels. The results indicate that a majority of the respondents' project teams (66.29%) have defined standards and guidelines. Participants reported the list of technologies used to develop the application software system (see Table 5.3 for the list).

Cool:Gen	Telon	DB2	CICS
Java	MS Visio	Test Director	.NET
SQL Server	Rational	Visio	Forte
COBOL	Holocentric	SmartCLient	Mainframe
M204	ETI	J2EE	Datastage
Metastage	XML	SAS	Visual studio
Arc GIS	ADO.NET	Powerbuilder	ASP.NET
Princeton	Platinum	Peoplesoft	JCL
VSAM	TIBCO	ORACLE	UNIX Shell scripting
MQ series	REXX	DWH	HTML
C#.NET	C	C++	Clearcase
SAP	AS400	RPG	CLP
SQL2000	Excel	Access	Web services
Siebel	Oracle apps	AVAYA	Business objects
Attunity	Ascential	Mercury	FileNET
Exchange server	Rational suite	Teradata	IMS
ERWIN	ETAL	IBEA	Apache
PHP			

Table 5.3. List of technologies

5.2.4 Duties and responsibilities for different roles

Table 5.4 shows the summary of data collected on duties and responsibilities for various roles from the general-survey. Table 5.5 shows the duties and responsibilities for other roles.

No.	Role	Duties and responsibilities
1.	Business analyst	Gather and analyse user requirements Write functional specifications Build a process model Testing Liaison with business team Liaison with technical team Write technical documentation Function testing Support testing Requirement knowledge User Interface design
2.	Programmer	Writing up design document Programming according to functional spec. Unit testing Developing database design. Review Monitoring production Production support for systems Ensuring standards are complied with. Defect fixing
3.	Programmer-Analyst	Analysis, Programming according to functional spec, unit testing design and writing up design document, review Production support for systems, developing web based applications Ensuring standards are complied with. Developing database design. Defect fixing, quality assurance, deployment, Mentoring other developers, Release applications, data fixes, release coordination, stakeholder management, post implementation monitoring ,Attending team meetings
4.	Project Manager	Manage a team, Manage project office, Business system management, Allocate work, Programme management of application development projects, Manage budget, issues, risks, Requirements management Testing management, Design management, Manage Schedule, cost and staffing
5.	Architect	Lead Architect and system administrator for documenting software. Responsible for production support, upgrades, customization to address business requirements, Understanding technical Architecture. Technical testing, software support and software development. master system maintenance, Development, Design and Implementation
6.	Designer	Analysis, functional design and mapping Design conversion programs, maintain associated checklists and procedures Gather requirements, impact analysis, design and offshore co-ordination, project management
7	Tester	Tester of user requests and system updates/fault corrections Function testing, writing test cases, running batch processing Testing mainframe applications, preparation of testing plans, testing strategies, scripts and documenting errors found during testing System testers mainly front-end and backend testing Validate and verify Billing Systems: Test Analyst

Table 5.4. Duties and responsibilities for various roles

No.	Role	Duties and responsibilities
1.	IT Security expert	Review security arrangements and provide advice.
2.	Systems developer	Configuration and support of software product and develop and update the software from time to time.
3.	Release manager	Release and Data Management
5	Tech support	Production monitoring and support, develop tool to assist with the support Mainframe support, application promotion and testing support, production support, automated test tool support, SQL Server DBA, Cool:Gen model management, VB App developer
6	System admin	Production support, Batch scheduling, help desk, support, RACF administration.
7	Database analyst and Database Administrator (DBA)	Database infrastructure, implementation and operation support, Database Administration, Performance Tuning and data conversion.

Table 5.5. Duties and responsibilities for other roles

5.2.5 Summary

The data for the general-survey was collected from a range of geographically distributed cities. The general-survey has gathered data from software professionals with varying experience, different levels and areas of qualifications and working at differing industry segments. This section summarised the demographic profile from general-survey data.

5.3 Analysis of Part B: Knowledge requirements

5.3.1 Introduction

The general-survey questionnaire collected data on the perceived knowledge needs to perform SE roles. This section summarises this data using descriptive statistics. The qualitative data collected is also categorised and presented in this section.

5.3.2 Quantitative results

Table 5.6 shows the means of knowledge factors identified in the theoretical framework required to perform various roles in SE from the general-survey data collected.

Knowledge Type	1-BA	2-Programmer	3-Programmer/ Analyst	4-Architect*	5-Designer	6-Tester	7-Project Manager
Business Domain Knowledge	3.04	2.64	2.91	3.00	3.40	2.80	3.54
Configuration Know.	2.79	2.98	3.24	3.14	3.03	3.10	3.03
Culture	2.65	2.38	2.38	2.13	2.83	2.23	2.96
heuristics	2.94	2.51	2.67	2.63	2.85	2.90	3.06
Process	2.91	2.93	2.96	2.67	3.40	2.40	3.08
Social network	2.65	2.49	2.52	2.78	2.87	2.73	2.76
Soft skills	3.20	2.92	2.97	3.20	3.24	3.35	3.45
Systems	3.20	2.72	3.01	2.89	3.27	2.93	2.99
Core Technique skills#	3.68	3.44	3.61	3.67	3.60	3.80	3.46

Table 5.6. Data on knowledge required for performing various roles*

*: Scale for the importance of a knowledge type to perform the job: 0-not applicable, 1-little help, 2-moderate help, 3-great, 4-necessary.
- Core Technique skills: The technique skill items that are directly relevant to perform the job for that role are termed as core technique skills. A criterion was applied to select the top skill items that have an average rating of ‘Great’ or above.

5.3.3 Qualitative results

The general-survey questionnaire, in Part-B, collected examples of knowledge required to perform SE roles. The qualitative data on examples of knowledge items collected together with their categorisation are presented in Table 5.7.

Role	Comments by participants on knowledge required to perform their job	Category (Type of Knowledge)
1-BusinessAnalyst	Knowledge of business processes. Example; if a customer makes an online booking, who /what/when is the next step etc.	Configuration knowledge
	Knowing individual’s (colleagues, stakeholders, the business) personality traits, background, skill sets, drivers, etc.	Social network knowledge
	Knowledge of external government standards, practices, requirements is a great help for enabling a web presence	Heuristics knowledge
	Knowledge about different tools, Enterprise Architecture	Technique skills
	Overview of architecture design to understanding of interfaces within B2B applications	Configuration knowledge
2-Programmer	Mostly browsing net for problem and implementing them in projects helped in completing jobs	Systems knowledge, Heuristics knowledge and Technique skills
	Programming skills and knowledge	Technique skills
	The relationship with senior management team is very important. It helps the juniors to learn a lot and gives a great scope to learn and know about the latest technologies or update of a current technology	social network knowledge
3-Programmer/Analyst	Very important: Knowledge of the language	Technique skills
	People skills	Soft skills
	Ability to share job/work load/knowledge with other team members	Soft skill
	Similar business domain area knowledge.	Business domain area knowledge
	Political environment/culture, power and individuals ability to influence my tasks	Cultural knowledge
	Ability to construct tools to automate repetitive tasks performed in monitoring production applications. Very important in improving effectiveness of staff involved in repeating/boring activities.	Technique skill
	People management	Soft skills
	Knowledge of soft skills/ personality skills will help greatly in job. Such as; Communication, discipline, health awareness, patience, perseverance, smart disposition and behaviour.	Soft skills
	Knowledge of other sub-systems which interact with other application systems	Configuration knowledge.
	Business knowledge and functional knowledge	business domain area knowledge
4-Architect	Development and software release Management	Technique skill.
	Business Added Analysis, Business process modelling	Technique skill.
	1) Understand operating systems, Troubleshooting skills and debugging techniques. 2) Researching on internet forums, Google searching.	Heuristics knowledge, systems knowledge and Technique skill.
6-Tester	SQL knowledge, experience in testing tools	Technique skill.
	Understanding different levels of expectations, both of customer and project necessary.	Cultural knowledge
7-ProjectManager	Data modelling concepts, Data Modelling software Tool	Technique skill.

Table 5.7. Qualitative data on knowledge required to perform various roles

5.3.4 Discussion

The general-survey sought the participants to indicate the extent to which various knowledge items helped them perform their role successfully. The statistical mean knowledge needs as seen in Table 5.6 and qualitative examples (see Table 5.7), support the proposition that knowledge factors identified in the theoretical framework have been required to perform for different roles in SE. Although, it appears that the level of knowledge thus required varies from role to role.

5.3.4.1 Reliability

The research proposed a theoretical framework between knowledge factors and performance (see section 3.5.2). The knowledge factors were operationalised into questionnaire items. Table 5.8 shows the calculated reliability of various factors. Cronbach's alpha value of 0.7 or above indicates reliability (Nunnaly, 1978). However, for exploratory purposes a lower limit is acceptable (Hair et al., 1998). While 1) Configuration knowledge, 2) Soft skills, 3) Cultural knowledge and 4) Heuristics knowledge show high reliability, Cronbach's alpha values are in the borderline for a) Technique skills (0.610), b) Social network knowledge (0.598) and c) Systems knowledge (0.617). However, for exploratory purposes these values are acceptable. A re-factoring using Exploratory Factor Analysis (EFA) can be carried out to un-cover any possible underlying classification in the general-survey data (EFA results reported in section 7.3).

	Factor in theoretical framework	Reliability Cronbach's alpha
1.	Soft skills	0.885
2.	Technique skills	0.610
3.	Business domain area knowledge	Not applicable due to only item on the questionnaire
4.	Knowledge of processes	Not applicable due to only item on the questionnaire
5.	Social network knowledge	0.598
6.	Systems knowledge	0.617
7.	Cultural knowledge	0.889
8.	Heuristics knowledge	0.847
9.	Configuration knowledge	0.875

Table 5.8. Cronbach's alpha for items based on original theoretical framework

5.3.5 Findings and conclusion

This part of the analysis of general-survey set out to identify the relevant types of operational knowledge required to perform various roles in SE teams. A theoretical framework has been developed and tested using a general-survey. The results support the proposition that the factors 1) cultural knowledge 2) configuration knowledge 3) soft skills 4) process knowledge

5) heuristic knowledge 6)social network knowledge 7) systems knowledge 8) Technique skills and 9) business domain area knowledge have been required to perform various roles in SE teams. The Cronbach's alpha of some of the factors (Technique skills, Social network knowledge and Systems knowledge) from the general-survey is border line for acceptable reliability. A re-factoring can be carried out to un-cover any possible underlying classification in the general-survey data.

5.4 `Analysis of Part C – Knowledge and Performance

5.4.1 Introduction

The second research question aims to identify the knowledge factors that contribute to performance improvement, since joining the team. The Part-C of the general-survey questionnaire captured the perceived contribution of different knowledge types to increase in performance. Report of analysis carried out in accordance with the plan, developed in section 4.4.6, is reported in this section. Qualitative data collected is categorised and summarised.

5.4.2 Qualitative results

The general-survey contained a question (Part-C; Identifier: C3; see full questionnaire in Appendix 11.3) to the participants to provide one significant example of how the increase in knowledge (of different types) contributed to their increased performance. Sixty one comments were obtained. These were classified based on identified knowledge factors (see Table 5.9 and Table 5.10). The analysis of qualitative data reveals support the proposition that the different types of knowledge listed in the theoretical framework contribute to increase in performance. The examples demonstrate support for the contribution of increase in knowledge (of different types) to improvements in performance and are summarised below:

- Business domain area knowledge: An understanding of business domain knowledge such as insurance particularly during requirements analysis can help in improved performance.
- Configuration knowledge: Overview and architecture of the system and its functionality of the system including business rules, technical components of the system, its database structures help in resolving problems, analyse impacts and in improved performance.
- Cultural knowledge: An understanding of cultural items through knowledge sharing sessions and interactions with senior management, an understanding of quality requirements for products, expectations from people on work outputs, impact to people have helped with increasing the performance.
- Heuristic knowledge: Management of templates and its use and implementing lessons learned from project management reviews have helped with improved performance
- Process knowledge: Better knowledge of auditing procedures, test processes and management, procedures for analysis has helped with improved performance.

- Social network knowledge: Improved team communication and knowledge sharing culture, expanded stakeholders, personal relationships, connecting people, closer working relationships, an understanding of point of contact helped improved performance. The following comment demonstrates the importance of social network knowledge to improved performance. *'Social Network knowledge – once I knew who did what within the organisation, ability to quickly resolve external problems which were affecting the application I supported improved dramatically.'*
- Technique skills: Improved technical skills in new or existing technologies contribute to improved performance.

It is not clear why participants failed to provide qualitative comments for 1) soft skills and 2) systems knowledge.

Business domain area knowledge	Configuration knowledge	Cultural knowledge
• Knowledge attained about accounts online helped me to understand the integrated nature of the modern system and the system being developed for the program. • Knowledge of insurance in previous company has helped greatly to understand the current project domain and greatly help to perform project tasks. • It makes where the existing system fits with the Business domain. Requirement impact analysis is very simple and will take less time and in term result in increased performance. • As I am almost involved in requirement analysis and design the knowledge of the current business domain helps me in finding out the architecture of the changes and to convert them into technical words so that the programmers can code that. • The project I am currently on I on-going in nature so the business knowledge I continue to gather improves my performance over time. As the project moves forward, it allows me more time to make decisions and perform design tasks, and less time tracking down key business/client personnel to get the answers I am looking for or need. • Knowledge of the Business Domain, along with Social Network, due to the length of time spent on the job with the same group of staff a customer is a significant factor in being able to perform efficiently • Understanding of Railroad business.	• My understanding of the system architecture allowed me to find solutions that better fit the architecture. • When supporting an application, it’s important for me to: 1) Have an overview of the system I am supporting, to see how various components fit together. 2) Have a good understanding of the business rules of the system. It would also help to have knowledge on single points of failure of the system. • Knowledge of existing functionality and processing has made it easier to understand the impact of required changes • Better understanding of Database structures lead to an improved problem solving ability in the areas of screen information problems, letter creation and problem impact assessment via SQL queries • Knowledge of the system/business process made it easier to analyse system change requests • Interference between my system and on internal system was a major issue. Increased understanding of the dependencies between systems, infrastructure/design of any system, and also increased networking in terms of understanding people who could assist has allowed these issues to be investigated and solutions put in place • The overall architecture, database structure, user interface, programme functionality of the application system • I’m developing a web enabled database. There is an existing database. It is helping me to understand the organisational activities • Comprehensive understanding of all technical aspects of the application, its objects, interactions and functionality. • Production support experience gives the main knowledge to increase any performance • You need a solid understanding of the technical components of the system to be able to properly leverage all of them to meet the requirements of the business • Especially data interchange between several railroads improved my performance • I have understood the data model of our system. This helps me to fix the production problems very fast.	• Exposure and increased understanding of quality products has resulted in higher expectations of what will be delivered and the resulting increase in the performances of the product • Hard work, team management support • Knowledge sharing sessions – an interaction with our senior management helped us a lot to increase our performance • Instances in relation to the management of under-performance of team members

Table 5.9. Qualitative data Part I: Knowledge contribution to performance

Heuristics knowledge	Process knowledge	Social network knowledge	Technique skills
• Ensuring that people use the correct document templates, changing the templates, as required, to enable them and to capture more aspects of the business requirements. • Project management lessons learned. You are able to make each project better and improve your performance with benchmarking of similar tasks	• Better knowledge of how staff work allowed auditing activity to be better collected and analysed • Test process and management • Knowledge of process/procedure contributes to the analysis of a system.	• Team communication greatly increased since I joined the team. This improvement has been invaluable in keeping the whole team informed on what each member is up to. It also fosters a knowledge sharing environment which helps to prevent costly duplication of effort • Recent shift from developer lead to business and developer lead has required expanding contacts to stake holders internal and external. • Friendly colleague or boss helps to improve performance and knowledge • Social network knowledge helps me daily to be able to organise time, multi task, reduce time to complete tasks and put the right people in touch with each other. My meetings can stay focused and on-time because I can bring the right people to the table and obtain agreement without later changes. • By establishing closer working relationships with business clients can better negotiate required compromises on system features that can be afforded with budget funds and resources • Our system interacts with multiple external systems. If any of the external system fails during production, I did not know whom to contact. Now, I know the external systems as well as the point of contact which helps me to solve the problem very fast. • Social Network knowledge – once I knew who did what within the organisation, ability to quickly resolve external problems which were affecting the application I supported improved dramatically.	• New technical skills show to perform programming tasks prove efficiency • Coding in C# for the last year has significantly increased my knowledge and programming skills • Increase in volume of work interface between the mainframe and servers. This requires a new knowledge skill set (which had to be learnt). • It was easy to move on and develop for the compact framework (mobile development) after having worked in the .NET framework for 1 year and more. After having worked in ASP.NET and Visual Studio .NET 2003 for a year or so it was easy to work and adapt to working in Visual Studio .NET 2005. • Improved knowledge of HTML CSS formatting. Allows improved architecture of web page design. • Initial work involved definition of HW/SW architecture. New tool environment for me, so had to learn how new tool works to define environment and move forward with. • Learning Perl and SQL commands • When I started, I had limited knowledge of the technical programming language, COOLGEN. This meant I had to continually ask other for advice. I now rarely need to do that and thus I am more efficient in programming. • My knowledge of Batch processing (technique) and business domain helped in improving the performance of the application by re-design, change in Application Architecture. The interface response time significantly improved by applying business knowledge • Joined current organisation as a software engineer but now I handle roles like Project Manager and Business Analyst (BA) and this knowledge came in handy.

Table 5.10. Qualitative data-Part II: Knowledge contribution to performance

5.4.3 Findings and conclusions

Section 4.4.6 outlined the plan by which the second research question on knowledge factors contributing to performance is investigated. Analysis is carried out based on this plan and the findings are presented. Part C of the questionnaire collected the following data elements.

- For every knowledge factor identified in the theoretical framework (nine types of knowledge; see Figure 3.7), perceptions of the individuals on the contribution of increased knowledge to improved performance is captured. (Question C.2.B.n; where n denotes the identifier for knowledge type; see Appendix 11.3 for general-survey questionnaire.)

The mean perceived contribution to improved performance for the nine knowledge factors identified in the theoretical framework was computed and ranking assigned (see Table 5.11). The ranking shows that increase in technique skills contributes most to improved performance followed by increase in configuration knowledge. This is followed by increases in systems knowledge, business domain knowledge, soft skills and social network knowledge respectively. Increase cultural knowledge contributes least to performance. Increases in Process knowledge and heuristics knowledge rank 7 and 8 respectively.

Rank#	Type of knowledge	Mean perceived contribution to improved performance*
1	Increase in Technique skills	2.15
2	Increase in Configuration knowledge	2.12
3	Increase in Systems knowledge	2.04
4	Increase in Business domain knowledge	2.01
5	Increase in Soft skills	2.00
6	Increase in Social network knowledge	1.95
7	Increase in Process knowledge	1.77
8	Increase in Heuristics knowledge	1.68
9	Increase in Cultural knowledge	1.66

- Rank based on mean contribution to improved performance

- Scale used 0 = nil, 1 = small, 2 = moderate, 3 = great

Table 5.11. Ranking: Mean perceived contributions of knowledge types

Based on the quantitative and qualitative results (see Table 5.11 and section 5.4.2), it can be concluded that all the knowledge factors identified in the theoretical framework contribute to improved performance. As per the rankings, the increase in technique skills contributed most to improved performance. Due to the technological nature of SE, the need for software skills has been identified since the birth of systems and programming and more emphasis provided

(Henry, 1974, cited in Benbasat et al., 1980). Technical skills are given importance and prominence in newspaper advertisements for recruitment when compared to contextual knowledge (Todd et al., 1995; Wade and Parent, 2001; Gallivan et al., 2004). Due to their core nature, it can be expected that technique skills such as software skills, project management skills and business analysis skills contribute most towards performance of individuals. Configuration knowledge, ranked second, is predominantly contextual to the organisation and its contribution to improved performance is in alignment with existing qualitative research reports (Curtis et al., 1988; Sawyer and Guinan, 1998). It should be noted that this research is an exploratory in nature and the first to rank knowledge factors contributing to improvements in performance.

5.4.4 Summary

This section reported the analysis of data on knowledge factors that contribute to improved performance. A ranking of based on the means of perceived contribution of knowledge has been developed. The ranking of knowledge factors based on their contribution to improved performance (see Table 5.11) shows that Technique skills (1^{st}) and Configuration knowledge (2^{nd}) rank the top, with Cultural knowledge (9^{th}) ranking at the bottom of the list.

5.5 ANOVA with demographic profile

The following statistically significant relationships were found between Total Professional Experience and contribution of various knowledge types to increase in performance (see Table 5.12). The detailed statistical evidence is available at Appendix 11.5.

- As the Total Professional Experience increases the perceived contribution of process knowledge to increase in performance declines ($p < 0.017$).
- As the Total Professional Experience increases the perceived contribution of general systems knowledge to increase in performance declines ($p < 0.033$).
- As the Total Professional Experience increases the perceived contribution of cultural knowledge to increase in performance declines ($p < 0.023$).

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.333	1	3.333	4.679	.032[a]
	Residual	109.676	154	.712		
	Total	113.008	155			

a. Predictors: (Constant), Total Professional Experience

b. Dependent Variable: C25B.Cntrbtn. of systems kno. to inc. in performance

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.583	1	3.583	5.317	.022[a]
	Residual	103.105	153	.674		
	Total	106.688	154			

a. Predictors: (Constant), Total Professional Experience

b. Dependent Variable: C27B.Cntrbtn. of cultural kno. to inc. in performance

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	4.358	1	4.358	5.892	.016[a]
	Residual	112.407	152	.740		
	Total	116.765	153			

a. Predictors: (Constant), Total Professional Experience

b. Dependent Variable: C24B.Cntrbtn. of process kno. to inc. in performance

Table 5.12. ANOVA results: Total Professional Experience

Total Professional Experience can be considered to be aligned to the age of the individual. We can expect that predictions for age be applicable to Total Professional Experience as well. However, the question of relation between age and job-performance is largely unresolved with mixed results reported in literature (Ng and Feldman, 2008). Dokko et al. (2009) found a negative direct relation between prior experience and job-performance in insurance organisations and call centres. Total Professional Experience includes prior experience as well. The negative relation found by this research between Total Professional Experience and contribution of knowledge factors appears to support the findings by Dokko et al. (2009). Further, apart from Total Professional Experience and knowledge types there could be other factors such as achievement needs, effort, self-esteem (Rasch and Tosi, 1992) that could be contributing to the found relationships with knowledge factors. Hence, it is difficult to interpret these relationships. Further research needs to be carried out to understand the reasons for these relationships. However, it is an important finding that will be of significant help to KM. Individuals with higher Total Professional Experience will need to be provided with training, motivation and encouragement to manage the use of process knowledge, cultural knowledge and general systems knowledge in performing their job.

The following relations exist between Experience within team and Perceived contribution of knowledge types to improvement in performance (see Table 5.13). Detailed statistical results are presented in Appendix 11.6.

- There is a relation between Experience within the project team and Perceived contribution of increase in technique skills to increase in performance ($p < 0.002$)
- There is a relation between Experience within the project team and Perceived contribution of increase in configuration knowledge to increase in performance ($p < 0.022$)

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	8.041	1	8.041	11.864	.001[a]
	Residual	107.760	159	.678		
	Total	115.801	160			

a. Predictors: (Constant), Experience in Software Engineering Team

b. Dependent Variable: C21B.Cntrbtn. of Tech.skills to inc. in performance

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.601	1	3.601	5.412	.021[a]
	Residual	105.142	158	.665		
	Total	108.744	159			

a. Predictors: (Constant), Experience in Software Engineering Team

b. Dependent Variable: C29B.Cntrbtn. of config. kno. to inc. in performance

Table 5.13. ANOVA results: Experience within SE team

These results lend further support to the finding that increase in both Technique skills and Configuration knowledge contribute to improvements in performance (section 5.4 concluded that both these factors are the top ranking knowledge types that contribute to improved performance). Hence, from a KM perspective, these two skills need to be given higher importance and managed better after an individual joins a project team in order to obtain improvements in performance.

It was found that Role has a statistically significant ($p < 0.030$) relation with Contribution of configuration knowledge to improvements in performance (see Table 5.14).

Dependent Variable		Sum of Squares	df	Mean Square	F	Sig.
C29B.Contribution of configuration. Knowledge. To inc. in performance	Contrast	23.071	21	1.099	1.757	.029
	Error	85.659	137	.625		

The F tests the effect of A6.Role. This test is based on the linearly independent pair wise comparisons among the estimated marginal means.

Table 5.14. Relation: Role and Config. knowledge to increase in performance

The *Contribution of configuration knowledge to increase in performance* is higher for programmers, programmer-analyst, architect and designers. The *Contribution of configuration knowledge to increase in performance* is comparatively lower for project managers, testers and business analysts. Such a variation in average contributions can be expected since differing responsibilities of different roles (ACS, 2008) can require varying knowledge of application systems. However, ANOVA of pair-wise comparisons revealed no significant relationships between specific roles and *Contribution of configuration knowledge to increase in performance* (see Table 5.15).

A6.Role	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
1-BusinessAnalyst	2.043	.164	1.719	2.368
2-Programmer	2.240	.158	1.928	2.552
3-Programmer/Analyst	2.203	.095	2.015	2.390
4-Architect	2.667	.455	1.767	3.566
5-Designer	2.200	.352	1.503	2.897
6-Tester	1.750	.394	.971	2.529
7-ProjectManager	1.769	.219	1.337	2.201

Table 5.15. Mean contribution: Config. knowledge to increase in performance

In conclusion, it is found that there is a statistically significant ($p < 0.030$) relation between Role and *Contribution of configuration knowledge to improved performance*. Further analysis of means reveal that the contribution of configuration knowledge to improved performance is the highest for architects. This is followed by both programmers, programmer-analysts, designers and business analysts. The contribution of configuration knowledge to improvements in performance is the lowest for testers and project managers. This evidence suggests that increase in configuration knowledge is related to significant improvements in performance and hence KM activities should target for programmers, programmer-analysts, and business analysts with respect to configuration knowledge. Thus, the above findings are helpful in formulating appropriate knowledge management strategy within the SE organisations.

5.6 Limitations

Several limitations exist in regard to the general-survey research. These are listed below.

- Perception only – This research collects the perceived knowledge requirements and perceived performance requirements for performing various SE roles. Such perceptions have been used in the literature (Nelson, 1991; Lee et al., 1995; Nansi and Bennett, 1998; Wade and Parent, 2001) to capture knowledge requirements and are found to be a practical way to measure knowledge needs. These perceptions on knowledge may or may not reflect reality.
- The inarticulable tacit component knowledge present within individuals cannot be captured using perception studies on individuals. This research may not have captured this tacit knowledge adequately.
- The sampling methodology used in this study is purposive-judgemental sampling approach. This is a non-probabilistic sampling methodology. Hence, the results of the study cannot be generalised to a wider population.
- Different positions with same role title, within different organisations (even within the same organisation) involve different requirements for knowledge and may involve substantial variation and overlap in job specifications. Any generalised theoretical framework cannot apply to a given position straightway. A knowledge needs analysis (Nelson, 1991) needs to be performed with the available theoretical framework and the knowledge requirements should be identified.
- The study collects opinions on knowledge requirements for performing various roles and performance increases. Based on the data collected the study arrives at the knowledge requirements for performing various SE roles. However, the perceived knowledge requirements from individuals could be different from the actual knowledge requirements used for the job. Also, there may have been an ideal set of knowledge requirements to perform these roles, which is not known or not practised by the professionals in the industry. The research has a limitation in that aspect since it only collects the perceived knowledge requirements of SE practitioners. Further, the general-survey also does not differentiate knowledge and skill requirements by rating the performance of the practitioners.

- The general-survey collects opinions at a certain point in time and reports the results. The knowledge requirements and performance increases can change in the future due to various reasons (example: technological change, changes to role definitions). The general-survey cannot predict future. Any requirements variations in future must be researched and reported at that time.

5.7 Chapter Summary

A total of 178 responses were collected for the general-survey. The general-survey has gathered data from software professionals with varying experience, different levels and areas of qualifications and industry segments.

Both quantitative and qualitative analysis was carried out on the resultant data. The first part of the analysis of the general-survey set out to identify the relevant types of operational knowledge required to perform various roles in SE teams. The data collected from the general-survey provided empirical support to support the theoretical framework. The reliability values for some of the factors were in the borderline suggesting that a re-classification exercise of the factors might be useful for uncovering any underlying structure of the data. Categorised summaries of qualitative data provides rich set of real life viewpoints as further evidence in answering the research questions.

The second part reported the analysis of data on knowledge factors that contribute to improved performance. Ranking of knowledge factors (see Table 5.11) based on their mean perceived contribution to performance shows that technique skills (1st) and configuration knowledge (2nd) contribute most to improved performance.

Finally, the ANOVA was carried out between demographic profile items perceived contribution of increase in knowledge to increase in performance. The following statistically significant ($p < 0.05$) relations were seen.

- It is found that as the Total Professional Experience increases, the perceived contribution of process knowledge, general systems knowledge and cultural knowledge to improvements in performance declines.
- As the Experience within SE team increases there is higher perceived contribution of increase in technique skills and configuration knowledge to increase in performance.
- There is a statistically significant relation between role and perceived contribution of increase in configuration knowledge to increase in performance. The contribution is higher for programmers, programmer-analyst, designers and architects. It is comparatively lower for project managers, testers and business analysts.

These findings allow us to carry out focussed knowledge management in SE teams.

6 Case study research findings

6.1 Introduction

A case study of a Software Engineering (SE) team was undertaken (see section 4.5 for case study design) with the aim of identifying the knowledge required to perform various roles and knowledge factors contributing to improved performance of individuals. The aim of the case study is to provide an account of the real life context and experience, in answering the research questions. A single embedded, positivistic, predominantly quantitative case study was conducted, in a software project team within a large organisation. This chapter is a report of the analysis of data collected from that study.

An initial project profile of the SE team was gathered by using a key informant (the researcher). This project profile captured organisational contextual information associated with different types of knowledge. For example, specific configuration knowledge associated with the SE team such as user interfaces, interfaces and functions were identified. The captured contextual information was then used to design the project-survey questionnaire. The project-survey questionnaire was then distributed to the case study team members and data collected. After the survey responses were collected, an interview was then organised with participants of the project. Descriptive statistics of the collected quantitative data and summarised qualitative evidence are triangulated to answer the research questions.

The organisation of this chapter is presented in Figure 6.1. Following this introduction, the case study project environmental context is described in section 6.2. This context includes the technical skills, application software items and social networks that exist in the case study project team context. Section 6.3 shows summary of participant profile. Section 6.4 reports the analysis of quantitative data collected with respect to the research question on knowledge requirements to perform various roles in the SE team. Further, summarised qualitative data is also presented. Section 6.5 presents the analysis both quantitative and qualitative data on knowledge factors that contribute to improved performance. The limitations of the case study research are summarised in section 6.6 and the experiences of the researcher are reflected upon in section 6.7. Finally, the chapter summary is presented in section 6.8.

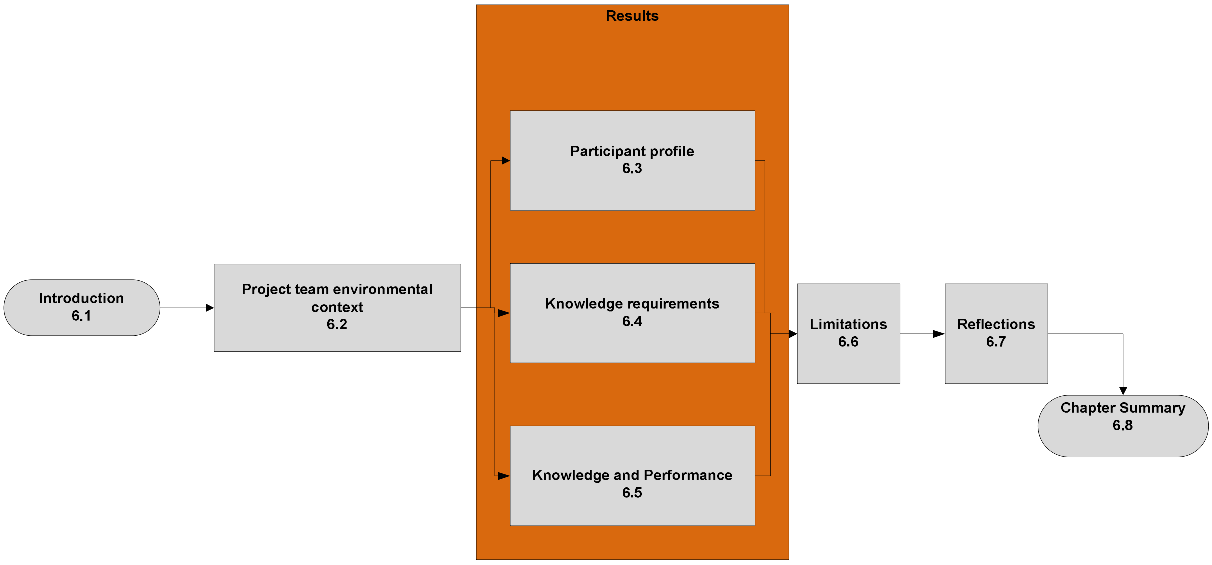

Figure 6.1. Flowchart: Organisation of Chapter 5

6.2 The case study project environmental context

6.2.1 Description of the Case study project team

The case study was carried out in a large Government organisation. The project team chosen works in the IT division of this organisation. The team mainly carried out enhancements to an application system, which supports case management activities. For the purpose of this report, the application system is referred to as Case Management System (CMS). The team consisted of Business Analyst (BA)-Testers, Programmer Analysts and two project managers. At the time the study was undertaken, the CMS application system had been in existence for six years.

6.2.2 Researcher as key informant

The author of this research also worked in this project team, as a senior consultant, at the time the research was conducted, for the last three and half years. No member of the team reported to the researcher. The key informant (the researcher) identified the following knowledge items as part of the initial project profile, by referring to documentation and other artefacts that were available. The project profile was validated with senior and knowledgeable staff in the project team.

6.2.3 Contextual knowledge items for use in project-questionnaire

During the operationalisation of factors into contextual knowledge items (section 4.3.2), it was envisaged that contextual knowledge types would be identified in the case study environment and used in the questionnaire. It was also envisaged that the key informant (the researcher), identify the contextual knowledge items based on various artefacts produced by the team and consultation with expert colleagues also within the team. For this purpose, senior team members with more than four years experience working on the application system and more than 10 years of total professional experience were considered as expert colleagues. The identified contextual knowledge items are presented in this section.

6.2.3.1 Technique skills identified

The list of technique skills required to perform various roles within the case study team can be identified by analysing the skills required to develop various artefacts and processes. The technique skills that are not predominantly contextual in nature, such as business analysis skills, testing skills and project management skills (see section 4.3.4.2 for operationalisation of technique skill items) were retained in the project-survey questionnaire. Downey and Power's (2007) suggestions on review of knowledge, skills and abilities required to create and use artefacts, was adapted to identify the software skills. The software item inventory was analysed and categorised by the key informant (the researcher) in discussion with expert colleagues. Being a transaction processing system, it can be categorised into Batch and Online (Stair and Reynolds, 2008). Reports and Interfaces with external systems were also categorised as either batch or online. However, development of reports required skills in a report development and generation tool. For each of the categorisations, an analysis of knowledge and skills and abilities to create and use the artefacts was made. The resulting software skills were validated with other existing documents, such as the selection criteria used for recruitment of programmer-analyst and business-analyst roles. Thus, the following software skill items were identified as required to develop the application system and included in the project-survey questionnaire.

- Application. Development Support Tool
- Application development Tool – Batch
- Application development Tool – Window
- Development methodologies
- Mainframe – ISPF
- Mainframe Platform used
- Relational Database Access Tools
- Report Generation Tool
- SQL

A review of documentation on architecture of the system and discussions with expert colleagues showed that the application system was developed using a form of radial architecture. Hence the item 'Knowledge of application software architectures' (as operationalised in section 4.3.4) was replaced by 'knowledge of application radial architecture' in the project-survey questionnaire.

6.2.3.2 Configuration knowledge

Configuration knowledge is defined as knowledge about the application software system. The system used a model based application system development process. Hence, it was possible to look at the repository for the application system and identify the list of configuration knowledge items. There were 16 processes from the CMS system which had interfaces with other systems. There were 103 user interfaces (windows). There were 54 batch processes and 30 reusable components. In the project-survey questionnaire, it is not possible to list all 203 software items since participants would lose interest in filling the project-survey questionnaire. Software items are selected based on trade-offs among factors such as software function, reuse plans, criticality, interface considerations, and other factors (Thawyer, 2004). Hence, in order to reduce the list, in consultation with expert colleagues, based on the identified trade-offs, items were grouped together to reduce the number of software items to 58. For example, the user interfaces relating to add, create, modify and list '*case*' were grouped into one item called *manage case*. Similarly, interfaces were grouped based on the systems with which interfaces happened. Following this process, a total of 58 software items were short listed for the project-survey questionnaire.

6.2.3.3 Social networks

It was envisaged during operationalisation (see section 4.3.4.5) that only the intra-organisational contextual social network knowledge would be identified and updated in the project-survey questionnaire. Also, it was envisaged that a full social network analysis would not be carried out due to ethical constraints. Based on discussions with members of the SE team, the finalised list of intra-organisational social networks included:

- Production support team
- Key interfacing system teams
- Enterprise architecture team.
- Environment support teams including DBA
- Batch run team
- Integration testing

- CMS Business team
- Release and change management teams

6.2.3.4 Cultural knowledge

Prior to the conduct of the case study research, a cultural mapping exercise[32] was carried out in the organisation. The documented results were made available to the key informant (the researcher). This cultural mapping exercise identified the elements of culture present within the IT organisation. These contextual items are included in the project-survey questionnaire in addition to items identified during operationalisation (see section 4.3.4.6).

- Understanding of leadership and vision at the organisation.
- Understanding of stories and myths in the organisation
- Understanding of power structures in the organisation
- Understanding of incentives (inducements such as bonus or awards for encouraging employees to perform better)
- Understanding of the way communication is carried out within the organisation
- Understanding of the way in which control systems work within the organisation
- Understanding of organisational structure
- Understanding of hierarchy (power associated with hierarchical ranks in the organisation)
- Understanding of rites and rituals in the organisation

6.2.4 Summary

The project-survey questionnaire is used to collect information from the case study participants. Thirteen members from a SE team in a major organisation participated in the case study research. Contextual information about the organisational context such as the application software items, cultural knowledge items, technical skills required to develop the application and social networks were collected using a key informant (the researcher). This information was updated in the project-survey questionnaire.

---

[32] based on focus groups and interviews conducted within the organisation by outside consultants.

6.3 Analysis of Part A: Participant profile

In total, thirteen members from the application software project team participated in the research. The following Table 6.1 and Table 6.2 show the profile of the research participants. Other demographic profile data are not shown in the thesis due to confidentiality reasons (see section 4.5.5 for accessibility conditions with the organisation).

Role	Number of participants
BA-Tester	5
Project Managers	3
Programmer-Analyst	4
Test and Implementation Manager	1
Total	13

Table 6.1. Number of participants and their roles

Qualification	Number of participants
High school certificate	3
Vocational certificate	3
Bachelors degree	5
Master's degree/Post graduate	1
Other	1
Total	13

Table 6.2. Educational level of participants in the case study

The researcher also worked as a consultant in the project team. Two members of the team, who completed the questionnaires, were not available for the interview process. Only summary data is presented in the thesis (see section 4.5.5 on accessibility conditions). As per the agreement with the case study organisation, the full case study research report was submitted to the organisation and was vetted and approved for release by the relevant manager.

To set the context of knowledge requirements to perform the roles, this section presents a summary[33] of responsibilities of the roles (based on question identifier: A.17; see section 11.3 for project-survey questionnaire). These are generally in alignment with the identified responsibilities from literature (see section 2.2.4.2).

Business analyst/Tester

Key duties and responsibilities of BA-Tester as gathered from the data are reported below.

[33] Duplicates removed from data collected and summary prepared.

- Analyse business requirements and present them in a form that can be used in designing the application system (write functional specification).
- Liaise with both business partners and developers to clarify any issues which may arise throughout the process.
- Prepare Test specifications- scripts
- Functional testing – Define the tests to be performed, carry out those tests and report the results of the test.
- Liaise with external testing areas of the organisation.

Programmer-Analyst

The following key duties and responsibilities of a programmer-analyst are identified from the project-survey data collected.

- Designing and coding online and batch changes for system release
- Mentoring and training new staff
- Estimating costs and resources required for system enhancements.
- Review functional specifications
- Involved in design of changes
- Review of technical specs and code
- Analysis, design and coding
- Documentation of programs and technical specifications
- Liaise with business analysts and stakeholders.

Project Manager

The following are key roles and responsibilities of the project manager identified from the qualitative data.

- Manage project team and manage project
- To ensure successful delivery on time within budget meeting client requirements.
- Negotiate scope, resources, funds, timeframe and outcomes/deliverables
- Manage process for each, to achieve outcomes agreed
- Assess budget, quality, HRM, satisfaction, client satisfaction and finalise/close project
- Observe Project Management principles.

Test and Implementation Manager

The following are key roles and responsibilities of the test and implementation manager as gathered from the qualitative data.

- Implement and manage a quality process for the planning, execution and reporting of the function testing effort.
- To ensure effective communication channels are maintained between the development team, test team and release management
- Work mostly on function testing, integration testing and implementation phases of the project.
- Carry out project management activities as well.

6.4 Analysis of Part B – Knowledge requirements

The case study research consisted of three phases of data collection. In the first phase, the case study project environmental context was gathered using a key informant (the researcher). This information was then updated in the project-survey questionnaire. In the second phase, responses were collected from participants using the project-survey questionnaire. In the third phase, further, interviews were conducted with participants to gather additional qualitative data.

This section presents the analysis of the quantitative and qualitative evidence that were collected to understand the knowledge requirements to perform various roles in SE. First, in sub-section 6.4.1, the summary of quantitative data using descriptive statistics is presented. The results show that the knowledge factors identified in the theoretical framework are needed to perform roles in the SE team. A discussion on the results, including qualitative examples gathered, is presented in sub-section 6.4.2. Finally, the findings and conclusions from this discussion are drawn in sub-section 6.4.3.

6.4.1 Quantitative results

The quantitative data collected from the project-survey was summarised using descriptive statistics. Table 6.3 shows the average knowledge requirements for identified knowledge types to perform different roles within the case study team. Table 6.3 shows that all the knowledge factors[34] identified in the theoretical framework are needed to perform various roles, at varying levels of need within the SE team. Configuration knowledge has been operationalised as the knowledge of application software items. The knowledge requirements levels denote the level of importance of a knowledge item to perform the job. It can be seen from Table 6.4 that the configuration knowledge is important for all different roles at varying levels. A programmer-analyst needs to have about 9.62% of the software items at a high knowledge requirements level. Business analyst/testers require 6.92% of the software items with high knowledge requirements level. Project managers need not know any software item at a high knowledge requirements level. This suggests that project managers need not have in

[34] Configuration knowledge is dealt with separately in Table 6.4.

depth knowledge of any software items. Thus, all roles require some level of configuration knowledge to perform their jobs.

Knowledge type	Business analyst/Tester	Programmer-analyst	Test and Implementation manager*	Project Manager
Core Technique skills$	3	2.4	2.5	3
Soft skills	2.45	1.8	2.29	2.26
Business domain area knowledge	2.2	2.2	3	2.5
Knowledge of processes and procedures	1.6	2	2	3
General systems knowledge	2.1	2.12	2	2
Social network knowledge	2.15	1.79	3	3
Cultural knowledge	1.38	0.95	1.68	1.9
Heuristics knowledge	1.6	1.94	2	1.63

Table 6.3. Averages requirements level for knowledge factors #*

* - Single embedded unit analysed.

\# Scale for the importance of a knowledge item to perform the job (knowledge requirements level) : 0-nil, 1-small, 2-moderate, 3-great.
$ - Core Technique skills : The technique skill items that are directly relevant to perform the job for that role are termed as core technique skills. A criterion was applied to select the top skill items that have an average 'Moderate' or above (value of 2 on the scale) (see note above on scale used).

Role	Software items with no knowledge requirement (%) (C1)	Software items with low knowledge requirement level (%) (C2)	Software items with medium knowledge requirement level (%) (C3)	Software items with high requirement level (%) (C4)
Project Manager	37.5	52.88	9.62	0
Programmer-analyst	48.08	21.63	20.67	9.62
Business analyst/tester	38.08	32.31	22.69	6.92
Test and implementation manager	94.23	1.92	0	3.85

Table 6.4. Configuration knowledge requirements for various roles

6.4.2 Discussion

Referring to Table 6.3 and chart in Figure 6.11, we can deduce that all the knowledge factors that were identified in the theoretical framework have varying levels of requirement to perform different roles in SE in the case study team. Thus the summarised data provides evidence to the theoretical framework proposed in this research. An analysis of the need for different knowledge types to perform various roles is presented in the following sub-sections. Qualitative data[35] collected from 1) the project-survey questionnaire (see Appendix 11.3 for questionnaire) and 2) the subsequent interview is quoted as part of the discussion.

[35] see section 4.5.3 for data collection design including interviews; Part-B of project-survey questionnaire and subsequent interview, sought comments and examples on knowledge required to perform job.

Soft skills

As we can see from Figure 6.2, the soft skill requirements for business analyst-tester are comparatively the highest and are very closely followed by the soft skills requirements for project manager. The Test and implementation manager also finds the need for soft skills at a higher level. The programmer analysts have a lower requirement for soft skills. It has been argued that programmer-analysts are more involved in creating code rather than interpersonal work and organising type work (Dollery, 2005) and the results from this study lend some credence to that suggestion. Ives and Olson (1981) showed that 76% of the time of systems managers is spent in oral interactions. It is found in literature that business analysis require soft skills such as interpersonal, coordination and communication skills (Evans, 2004). Thus, the results further demonstrate findings in literature. Several of the participants opined that the list of soft skills in the questionnaire was comprehensive. In addition, following additional examples of soft skills were provided by the case study participants.

- Skill to mentor new staff
- Ability to conduct good reviews on technical specifications
- Ability to transfer skills to another person in the team.

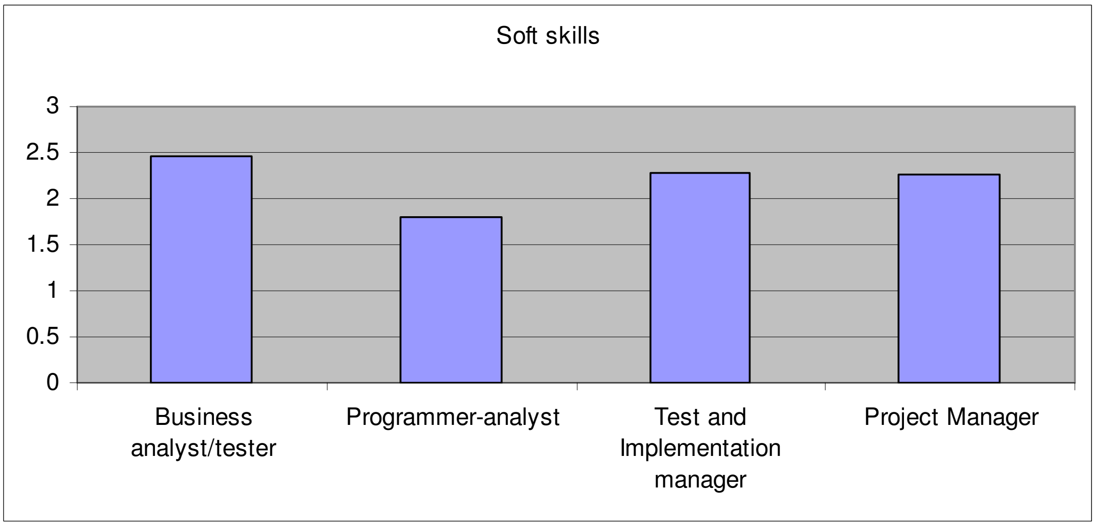

Figure 6.2. Soft skills for various roles

In conclusion, while all SE roles require soft skills to perform their job, project managers and business-analysts require comparatively higher levels of soft skills than programmer-analysts.

Technique skills

It is found that technique skills are required to perform various SE roles. A participant gave the example of 'ability to analyse dumps' as a technique skill required for programmer-analysts. The following three examples of technique skill items were provided by case study business analyst-testers:

- Writing queries using Structured Query Language (SQL),
- Running a Job Control Language (JCL) script and checking its output
- Ability to look at database tables.

The technique skill items that are directly relevant to perform the job for that role are termed as core technique skills. For example, a project manager's core technique skill is project management. As can be expected, the core technique skills requirements for each role are different. This expectation is reiterated in the collected data as well. Hence, project management skill has obtained higher ratings than the software skills with respect to performing the role of a project manager. On the other hand a programmer analyst needs higher level of software skills to develop the application but much less project management skills. Hence, it is not possible to compare the technique skills for different roles based on the chart shown in Figure 6.3. However, from this chart, it is possible to say that all the different roles require technique skills at high levels. Criteria were applied to select the top skill items that have an average of more than the rating of 'moderate'[36]. As can be expected, the results (see Table 6.5) show that core technique skills are related to their role and responsibilities.

Role	Knowledge items that form part of core technique skills
BA-Tester	Business analysis and Testing skills
Programmer Analyst	Software skills (Technical skills required to develop the application)
Project Manager	Project Management skills

Table 6.5. Core technique skills for different roles

[36] value of 2 on the scale. Scale for the importance of a knowledge item to perform the job 0-nil, 1-small, 2-moderate, 3-great.

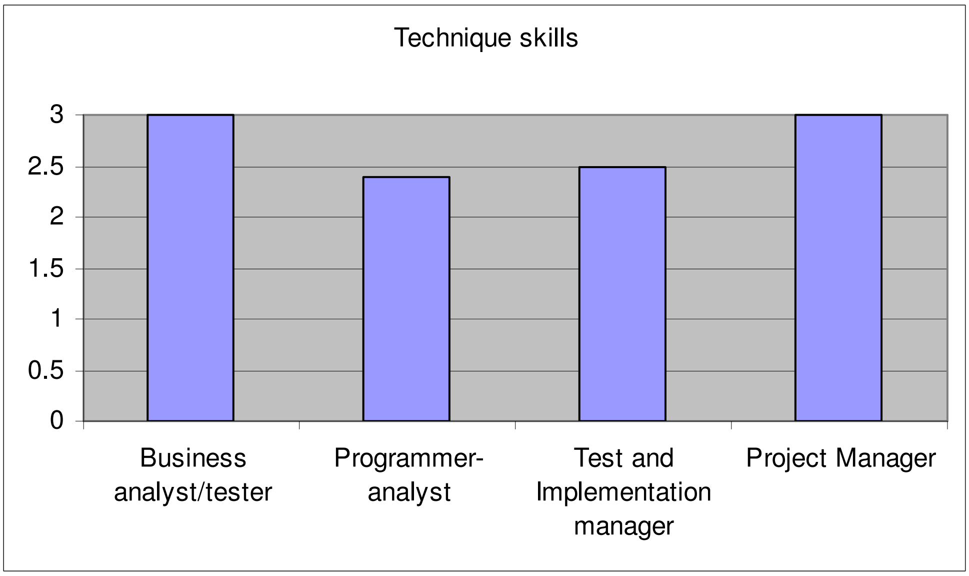

Figure 6.3. Core technique skills for various roles

Thus, it can be concluded that technique skills are required to perform various roles in SE but it appears that different roles require different kinds of core technique skills. Business analyst-testers require business analysis and testing skills. A programmer-analyst requires technical skills to develop the application. Project managers require project management skills and business analysis skills.

Process knowledge

Business analysts express less of a need for process knowledge than the other roles considered (Figure 6.4). But, project managers need to know the process to the highest level. This may be due to their role in ensuring that processes are followed. Programmers have to know a reasonable level of process knowledge to carry out their functions.

Examples of process knowledge items for business analyst-testers are

- Process followed for testing
- Test preparation phase
- Update parameters
- Run batch
- Test checking phase
- Document management processes.
- Some knowledge of System Delivery Methodology and CMMI

Examples of process knowledge items needed to perform the job of a programmer-analyst are as follows:

- Preparation of change specifications
- Code review including formal and informal reviews.
- Functional specification reviews with business
- Weekly status updates.

Examples of process knowledge items required to perform the role of a project manager are as follows:

- Understanding of corporate employee agreements including rules for leave
- Knowledge of processes relating to contracts with contract employees.
- Knowledge of rules and regulatory policies and procedures in administration
- Corporate system delivery processes and phases.

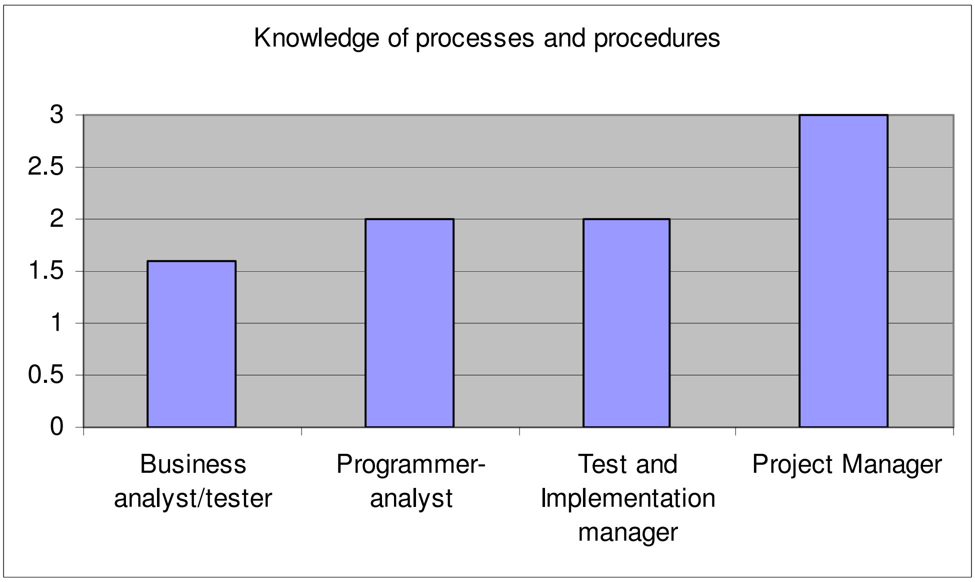

Figure 6.4. Process knowledge for various roles

Thus, it found that process knowledge is of importance to perform various roles in SE. It appears from the case study that the project manager requires more of this knowledge of processes and procedures than any other role.

Systems knowledge

All the roles require systems knowledge to perform their job (see Figure 6.5). Systems knowledge is an understanding of the interplay of cause-and-effect relationships that is essential for sound decision making relating to the job (Beazley et al., 2002). It includes skills such as problem solving and analytical and modelling skills. It can be seen that there is no real difference in systems knowledge requirement between different roles. It is possible that since all roles require sound decision making abilities, all of them require at similar levels.

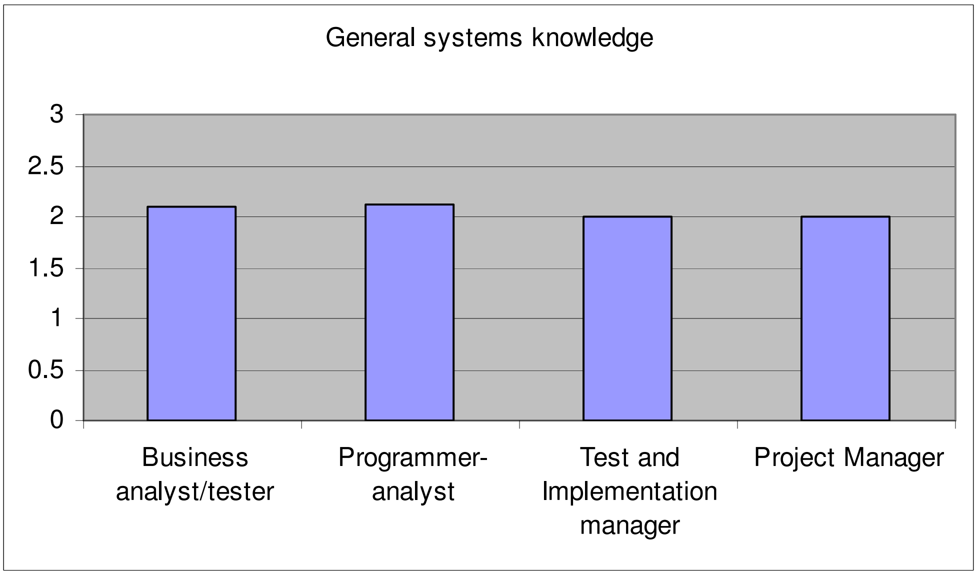

Figure 6.5. General systems knowledge for various roles

Cultural knowledge

Cultural knowledge was found to be of importance to perform various roles at varying levels. As we can see in Figure 6.6, project managers require higher levels of culture to perform their role. Project managers have a leadership role to play (Simerson and Venn, 2007). Leaders create culture and culture creates leaders (Schein, 2004). Hence, it appears that, in order to perform this leadership role, project managers need to know more about the culture of the organisation than other roles (see Figure 6.6). The following examples and scenario emphasise the need for cultural knowledge in performing various SE roles.

- The operational users who moved into the ICT business line emphasised the importance of cultural differences between operations and Information and Communication Technology (ICT) divisions
- The need for cultural knowledge is emphasised by the following comments:

"From IT culture point of view, the organisation has moved from a mindset that said 'allowing data access through internet is a serious security risk' to one of 'need to allow data access to improve service levels. The security risk can be managed. It is important for us to know the risk taking nature of the organisation"

- The need for knowledge of corporate practice statements is emphasised in the following comments.

"Corporate practice statements help shape the culture of the organisation Examples: IT usage guidelines, ethical guidelines."

- An understanding of APS values and code of conduct are required to perform the job.

Programmer-analysts need the least knowledge of the culture of the organisation. This is reinforced by a qualitative comment from one of the participant programmer-analyst: 'don't need to know the culture'. Business analyst-testers need to know culture only at a moderate level. The test and implementation manager rated cultural knowledge almost as high as the project manager.

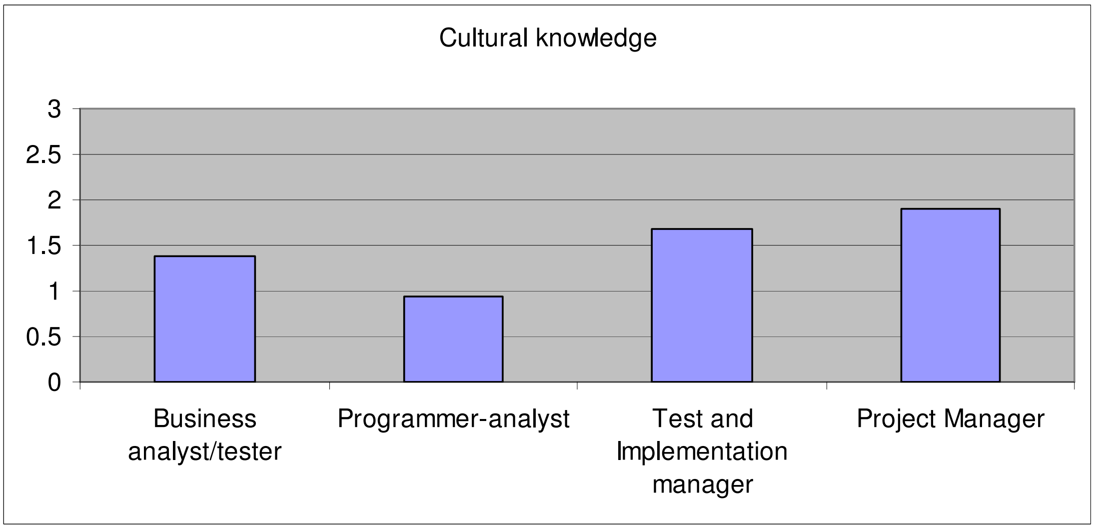

Figure 6.6. Cultural knowledge for various roles

Thus, the case study evidence shows that cultural knowledge is important for all roles. However, project managers require cultural knowledge at the highest level when compared to business analyst/testers. Programmer-analysts require the least cultural knowledge of all the roles

Heuristics knowledge

The chart (Figure 6.7) shows that heuristic knowledge is required to perform SE roles. Heuristic knowledge such as guidelines, shortcuts, rules of thumb and standards could be different for different roles and can vary. Examples for heuristics knowledge items for business analyst-testers are as follows:

- Level of detail expected of the testing efforts.
- Need for a clean run during testing. (A cycle of testing that goes with out any issues or problems is referred to as a clean run)
- When you work on a testing scenario you use the following thumb rule: "begin with an end in mind and work back how you do it"
- Follow the guidelines for testing
- 'Don't work harder; work smarter'
- Standards for functional specifications, window design, security standards.
- Templates

Examples of heuristics knowledge items for programmer-analysts are as follows:

- Previous experience using other languages
- Impact of coding on performance
- Organisational standards for application development and coding.
- Coding standards
- Object tracking
- Batch job scheduling and operational standards
- SearchWizard
- Version Analyser
- Naming of objects
- Test harnesses
- Organisational user interface standards
- naming standards
- Standard for public type components.
- Radial architecture standards

The following examples of heuristics knowledge items were captured during the interviews with project managers.

- Using templates developed
- Screen development guidelines, templates

- Costing and estimates – uses his own rule of thumb

The heuristics used by a programmer-analyst (Riel, 1996) in development are different for a project manager who may need heuristics in estimating effort (Hihn and Habib-agahi, 1991). The examples also appear to demonstrate differences in heuristics knowledge items. Hence, no comparison between the roles is made based on the case study results. In conclusion, the case research finds that heuristic knowledge is required to perform SE roles.

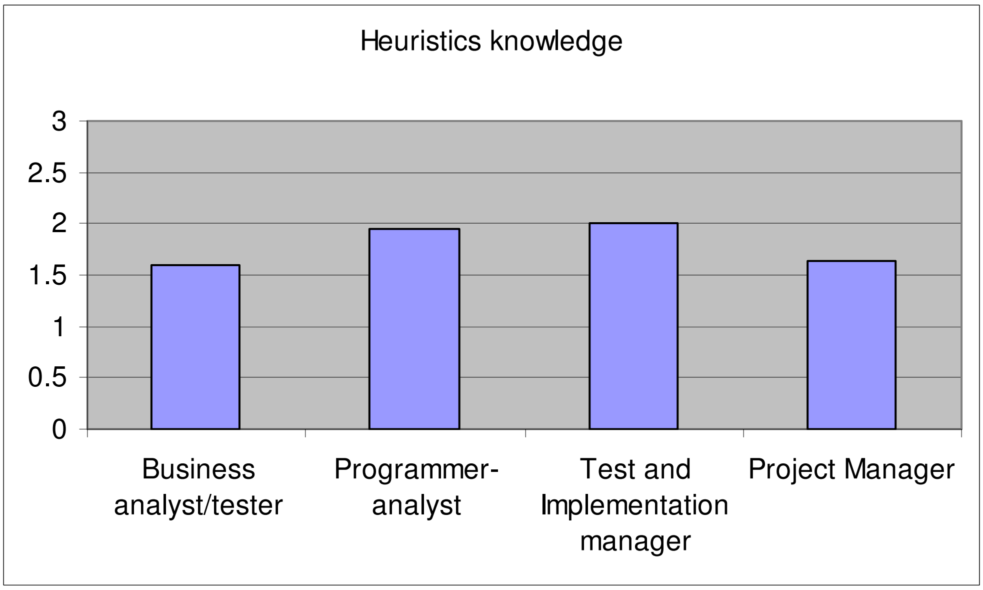

Figure 6.7. Heuristics knowledge for various roles

Social network knowledge

Social network knowledge is an understanding of the crucial relationships including the structure, interpersonal dynamics within the structure, common context and language held by individuals, that are required to perform the job (Lesser, 2000; Beazley et al., 2002). The following are comments illustrate the importance of the social network knowledge within the team, within the organisation and outside the organisation.

> *"Here information is available from people. If my mentor is not available, I approach another person who can help. "*

> *"If I don't know something, I research that part of life cycle. I know the rest of CMS. I may not know that part alone. There will be a learning curve to find out. But I don't*

think that will be too difficult. A couple of weeks on the job should do it. I will check references, manuals, talk to people – Not a huge issue"

"Identifying and finding documents required me to network with various people from Production Support Areas, Business and the team. In order to understand the processes, I had to look for documentation and talk to various people. The team had lot of knowledge. I relied on team members relating to processes and procedures followed before "

The following comments illustrate the need for social network knowledge for programmer-analysts.

"If I did not know about application radial architecture, and needed it in his job then I would approach someone who knows it straightaway."

The participant who made the comment pointed to another technical lead within the project team to whom he would turn into when needed. This statement shows the importance of social network knowledge to his job.

During the interview, one of the participants emphasised the following:

'Ring Fred for help if we don't know'.

The delivery of this statement showed that Fred, the business contact person is in regular contact with the participant. Note that Fred is in another team but within the organisation. The participant knows what Fred knows and can ask him for help as and when required. This statement shows the importance of social network knowledge.

The following scenarios extracted from interviews demonstrate the need for social network knowledge for project managers. These examples highlight that the Project managers not only need to know what each person in their team know they also need to know other stakeholders in the organisation.

"As a project manager, I need to know in detail what each of the team members know. Just knowing people as a person and know how they contribute to achieving the goals is essential to perform my job. If I ask myself whether this person will be able to do it or not ? "

"Social network is essential to perform my job. Understanding executives and other project managers and their interests helps. Using this one can influence other Project Managers. If a project or activity is considered not relevant by executives or project managers, it dies. Hence, it is important to know the priorities and interests of executives and project managers. Get everyone on side. Good relationship management within team and outside the team is essential to get that outcome. Managing relationship with Integration Testing team based on trust. "

The following comment illustrates the importance of social network knowledge outside the organisation.

"There are occasions when people don't know the answer. Once we didn't have the expert person. I networked through my wife to find the right expert person".

One of the participants has contacted people outside the organisation whom he considers as experts in various technical areas. He has talked to IBM-DB2 experts to see if they knew of answers to any queries. This illustrates the importance of social network knowledge outside the organisation.

Figure 6.8 shows the level of need for social network knowledge for various roles. The project managers and the Test and implementation managers have a high level of social network knowledge requirement to perform their job. Programmer-analysts, on the other hand, have the least amount of social network knowledge requirement, while BA-testers are in the middle. The project manager needs to build and establish social networks (Ives and Olson, 1981) and hence is expected to rely on social network knowledge. Programmer analysts need lower level of social network knowledge than business analyst/testers. This reinforces Dollery's (2005) suggestions on programmers being focussed on the tasks of programming to the exclusion of networking with people.

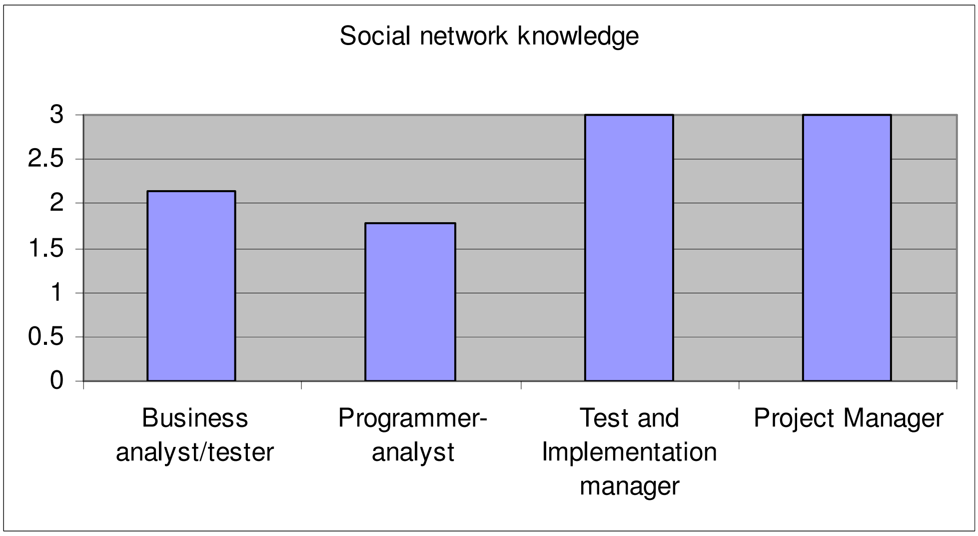

Figure 6.8. Social network knowledge for various roles

Thus, the case study finds that social network knowledge is required to perform various roles in SE. While, project managers require the highest, programmer-analysts require lowest level of social networking knowledge. Business analyst-testers fall in between these two roles in the need for social network knowledge.

Business domain area knowledge

The case study evidence shows that project managers require higher levels of business domain area knowledge than business analyst/testers. Perhaps due to the common business domain area involved in the application system, no qualitative examples were provided by participants. Both business analyst/testers and programmer analysts have almost similar need to have the knowledge of business domain area knowledge (see Figure 6.9). It was expected that the business analyst/testers would need the highest levels of business domain knowledge. However, the study finds that project managers require this knowledge more than business analysts.

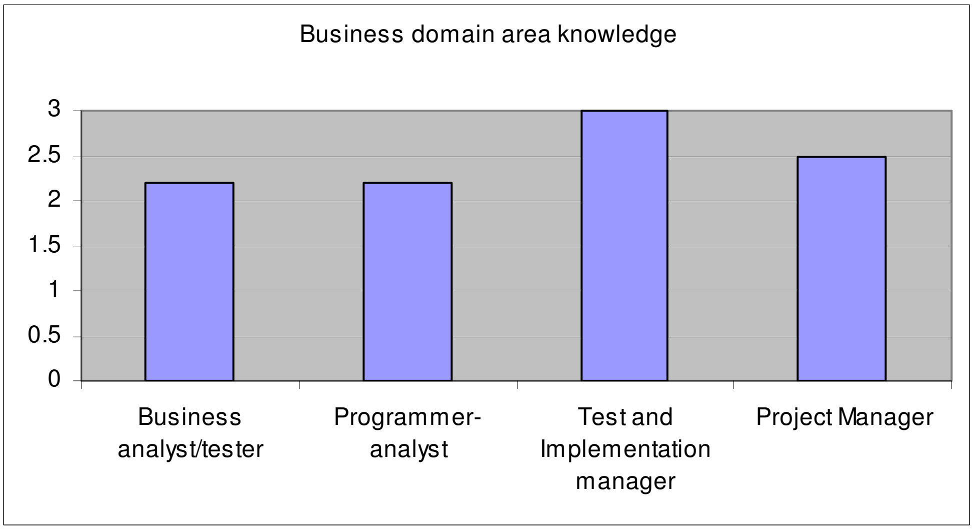

Figure 6.9. Business domain area knowledge for various roles

In conclusion, the case research finds that all roles require business domain knowledge with project managers requiring more of this knowledge than other roles.

Configuration knowledge

Configuration knowledge is operationalised as the knowledge of application software items. In the case study, the software items can be classified into following types: 1) batch processes, 2) online processes, 3) interfaces and 4) database table structures. All the participants indicated that an understanding of the overall architecture of the application and its database table structures are required to perform their job (see Table 6.4) . Figure 6.10 shows the percentage of items with different knowledge requirements levels required to perform different roles. Reviewing the various project-survey responses and interview comments, it appears that the specific set of knowledge items that were required at high levels depended on the work performed by the individual and included software items of different types.

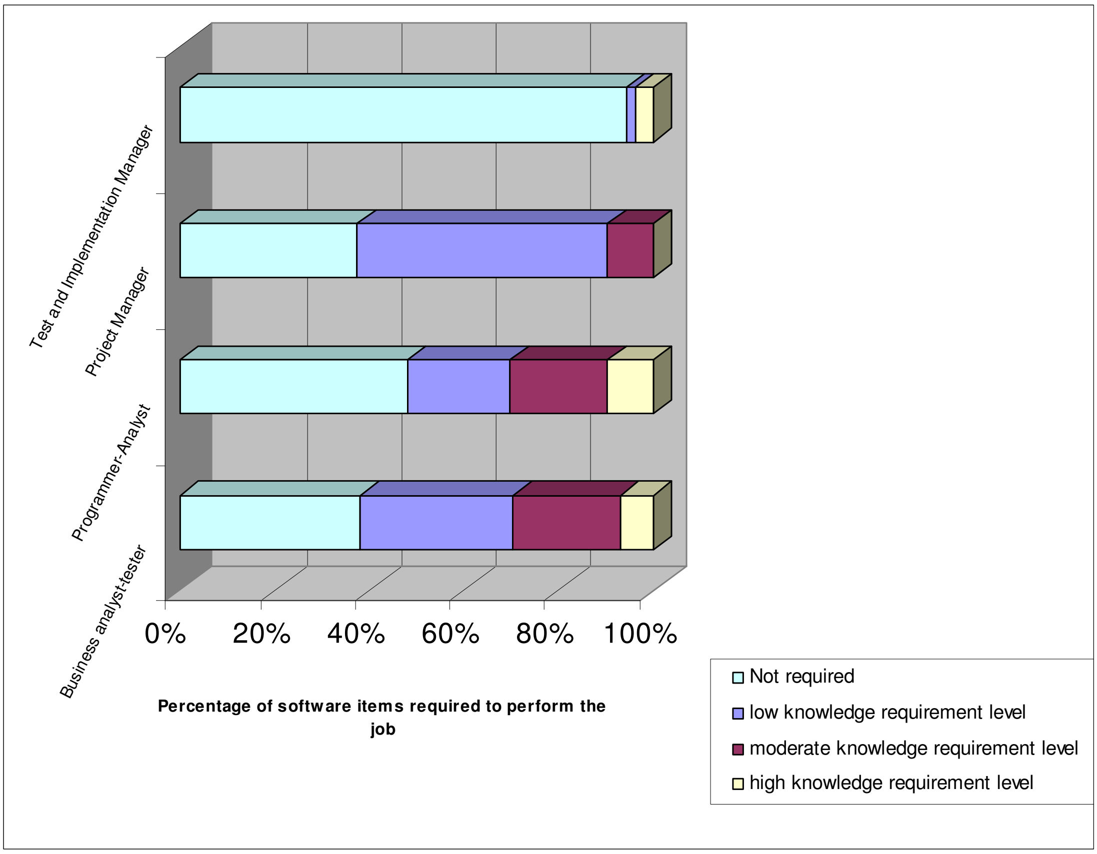

Figure 6.10. Configuration knowledge requirements levels for various roles.

As can be seen from Figure 6.10, project managers need to have a broad understanding of application software items. They don't need to possess deep knowledge of any software item. This is reinforced by the comment from one of the project managers as follows:

> *"In detail knowledge is not required. I can still do the job even if I don't know. If you know, you can ask the right questions, steer the team in the right direction"*

On the other hand, both business analyst/testers (6.92%) and programmer-analyst (9.62%) need to know some of the application software items to the detail. As the data points out, programmer-analysts need to know more software items in depth than the business analyst/testers. Further, programmer-analysts and business analyst/testers also require a broad understanding of the application software system.

It should be noted that there was no individual within the team who possessed an understanding of all the software items of the application. This emphasises the importance of team collaboration (Rosson, 1996; Barthelmess and Anderson, 2002; Cubranic et al., 2004; He, 2004; Ye, 2006) in SE.

In conclusion, configuration knowledge is required to perform all roles in SE. However, project managers only need to have a broad understanding of the application software system. Programmer-analysts and business analyst/tester require in depth knowledge of some of the software items and a broad understanding of others. Not one individual possessed the entire knowledge of the application system.

6.4.3 Findings and conclusions

Based on the knowledge requirements profiles developed for various roles (as seen in Figure 6.10 for configuration knowledge and Figure 6.11 for all other types of knowledge) and the qualitative information that has been summarised (in section 6.4), it is concluded that knowledge factors identified in the theoretical framework (see section 3.5) are important to performing various roles in software development at varying levels. Table 6.6 lists the findings for various knowledge types in the case study context based on the discussion from section 6.4.2. Thus, the results of the case study research provide empirical evidence to the theoretical framework proposed in this thesis. KM strategies[37] need to be developed based on an understanding of knowledge needed to perform SE roles. The results from this case study enable development of such KM strategies.

[37] examples: competency management and expert locator tools

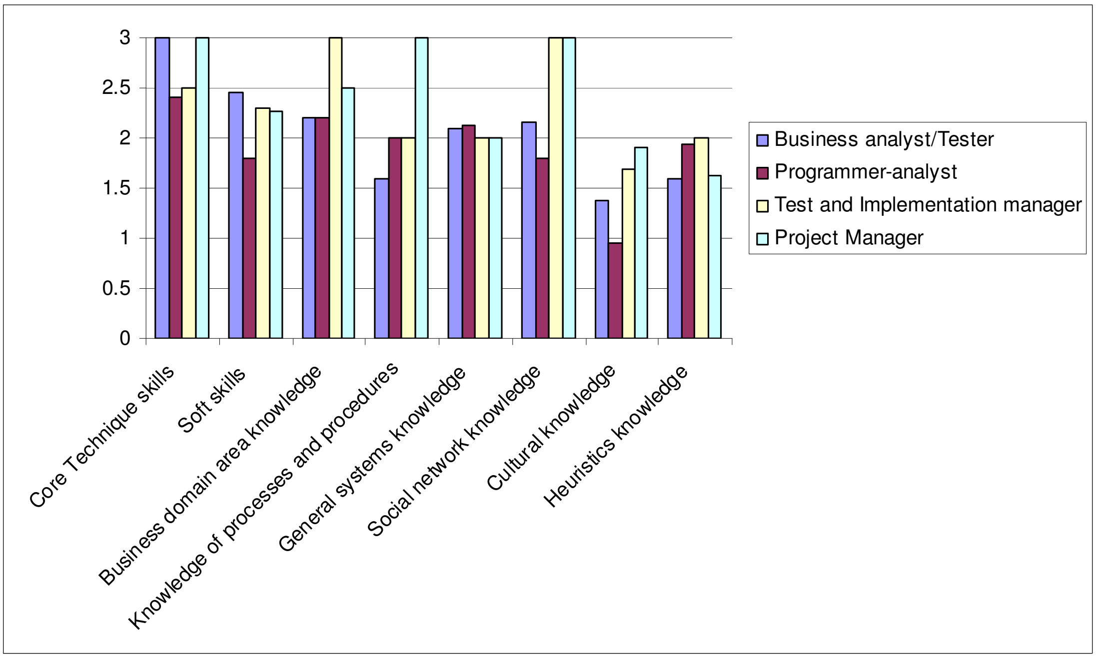

Figure 6.11. Comparisons for different roles #1,#2

#1: Scale for the importance of a knowledge item to perform the job: 0-nil, 1-small, 2-moderate, and 3-great

#2 – Configuration knowledge not included in the chart. It is dealt with separately.

No.	Type of knowledge	Finding from case study context
1.	Soft skills	Soft skills are required to perform different roles. Project managers and business-analysts require higher levels of soft skills than programmer-analysts.
2.	Configuration knowledge	Configuration knowledge is required to perform different roles with some requiring broader coverage and others requiring detailed coverage
3.	Business domain knowledge	The case research finds that all roles require business domain knowledge with project managers requiring more of this knowledge than other roles.
4.	Social network knowledge	The case study finds that social network knowledge is required to perform various roles in SE. Project managers require the highest levels of social network knowledge. Programmer-analysts require a lower level of social networking knowledge. Business analyst-testers fall in between these roles in the need for social network knowledge.
5.	Systems knowledge	Based on the case study data collected and analysed, we observe that systems knowledge is important in performing various roles in SE.
6.	Process knowledge	The process knowledge is of importance to perform various roles in SE. It appears from the case study that the project manager requires more of this knowledge of processes and procedures than any other roles.
7.	Technique skills	It is found in the case study that technique skills are required to perform various roles in SE. Different roles require different kinds of technique skills. Business analyst-testers require business analysis and testing skills. A programmer-analyst requires technical skills to develop the application. Project managers require project management skills and business analysis skills.
8.	Cultural knowledge	The case study evidence suggests that cultural knowledge is important for all roles. However, project managers require cultural knowledge at the highest level when compared to business analyst/testers. Programmer-analysts require the least cultural knowledge of all the roles.
9.	Heuristics knowledge	The case research finds that heuristic knowledge is required to perform SE roles.

Table 6.6. Findings: Knowledge requirements

These findings will help in formulating KM strategies. As suggested by Rus and Lindvall (2002), based on this finding, it is possible to formulate KM strategies to manage different types of knowledge for different roles in software project teams.

The results of the case study research appear to show that the knowledge and skill requirement to perform various roles reflects the responsibilities of the role. For example, project managers need to organise and hence network with people. This means, they need to possess higher levels of social network knowledge. Reflecting on their responsibilities, programmer-analysts need to possess higher software skills since they do the detailed design and coding of the software knowledge. Project managers and business analyst-testers require comparatively higher levels of soft skills than programmer-analysts. While configuration knowledge is required to perform all roles in SE, project managers appear to need a broad understanding of the application software system. Programmer-analysts and business analyst/tester require in depth knowledge of some of the software items and a broad understanding of others. Not one individual in the team possessed the entire knowledge of the application system. It is found that cultural knowledge, process knowledge, heuristics knowledge and systems knowledge are important to performing SE roles.

It is found from the case study research that technique skills are required to perform various roles in SE. Different roles require different kinds of technique skills. Business analyst-testers require business analysis and testing skills. A programmer-analyst requires technical skills to develop the application. Project managers require project management skills and business analysis skills. In summary, the case study research provides evidence that the knowledge factors identified in the theoretical framework have been found to be of need to perform various roles in SE.

Knowledge needs analysis is the process by which knowledge required to perform roles is identified. The findings from this research enable development and implementation of knowledge needs analysis to carry out competency management (Nelson, 1991; Davenport, 1997; Curtis et al., 2003). For example, a project manager needs a broad understanding of the application software system and hence can be trained on the configuration knowledge. Similarly, all roles require social network knowledge. Hence, KM strategies can be developed to gather information on the social networks relevant to the project team and train individuals to arm with that knowledge. In times of knowledge continuity management (Beazley et al., 2002), transition processes can explicitly include different types of knowledge transfer discussions so that the incoming person learns from the outgoing person. Educational curriculum can be reviewed to ensure that all the identified knowledge types are imparted to students reducing gaps between education and knowledge required to perform SE roles (Lee et al., 1995), thus considering the experiences of practitioners (Lethbridge, 2000).

6.5 Analysis of Part C – Knowledge and Performance

The second research question aims to identify the knowledge factors that contribute to performance improvement since joining the team. In order to answer this research question, the project-survey collected information about perceptions on 1) increase in knowledge and 2) improvements in performance since joining the team. This section reports the findings of the project-survey of case study participants. The section 6.5.1 summarises the quantitative results. Discussion on the results is provided in section 6.5.2. The findings and conclusions drawn are presented in section 6.5.3.

6.5.1 Quantitative results

For every operational knowledge factor identified in the theoretical framework, the project-survey collected the following data items:

- Perceived level of initial knowledge at the time the individual joined the team
- Perceived level of current[38] knowledge
- Perceived contribution to improvement in performance since joining the project team.

Based on the analysis plan (see section 4.5.4), the initial knowledge and current knowledge were converted to knowledge indices (unweighted normalised percentage; using the concept of knowledge index as described in Appendix 11.1). The increase in knowledge index was computed by subtracting initial knowledge from current knowledge. The average perceived contribution to increase in performance is also computed. The Table 6.7 shows the summary of mean and standard deviation for the 1) contribution to increase in performance and 2) increase in knowledge for different knowledge factors. Ranking is assigned to different factors based on its contribution to improvements in performance.

---

[38] current at the time the case study research was conducted;

Factor		Increase in Knowledge index (%)	Contribution to increase in performance#	Rank*
Configuration knowledge	Mean	25.23	2.23	1
	Std. Deviation	23.66	1.01	
Increase in social network knowledge	Mean	34.04	2.23	1
	Std. Deviation	22.43	0.93	
Increase in process knowledge	Mean	30.77	2.00	3
	Std. Deviation	25.32	1.00	
Increase in systems knowledge	Mean	5.13	2.00	3
	Std. Deviation	10.51	1.00	
Increase in technique skills	Mean	7.30	2.00	3
	Std. Deviation	12.51	1.01	
Increase in business domain area knowledge*	Mean	5.13	1.92	6
	Std. Deviation	12.52	1.04	
Increase in soft skills	Mean	1.17	1.85	7
	Std. Deviation	3.02	0.99	
Achievement needs	Mean	0.00	1.75	7
	Std. Deviation	0.00	0.50	
Effort	Mean	0.00	1.75	7
	Std. Deviation	5.52	1.15	
Increase in heuristics knowledge	Mean	14.10	1.62	10
	Std. Deviation	16.10	0.87	
Increase in cultural knowledge	Mean	15.06	1.31	11
	Std. Deviation	16.73	0.95	
Self-esteem	Mean	0.00	1.20	12
	Std. Deviation	0.00	1.10	
Personal reasons	Mean	0.00	0.80	13
	Std. Deviation	0.00	0.84	

Table 6.7. Summary tabulation for case study results

- Scale for contribution to increase in performance 0 – nil, 1 – small, 2 – moderate, 3 – great

* - Ranking based on contribution to increase in performance

6.5.2 Discussion

On average, participants have reported increases in all knowledge types since joining the team (see Table 6.7). Further, Figure 6.12 shows that all identified factors contribute[39] to corresponding increase in performance. It should be noted here that the project-survey directly captured the perceptions of participants on contribution of identified factors to improvements

[39] the average contribution is greater than zero; not through statistical relations.

in performance. The ranking of various types of knowledge based on their perceived contribution to increase in performance (see Table 6.7) shows that in the case study project team, configuration knowledge and social network knowledge contribute most towards increase in performance. Process knowledge, systems knowledge and technique skills rank joint 3rd in their contribution to increase in performance. The results of the case study support the findings of Rasch and Tosi (1992) that both 1) dispositional factors and 2) Knowledge, Skills and Abilities (KSA) contribute to improved performance. The standard deviation of contribution to improved performance ranges from 0.5 to 1.15 (see Table 6.7). Thus, the data collected from the case study provides support to the theoretical framework.

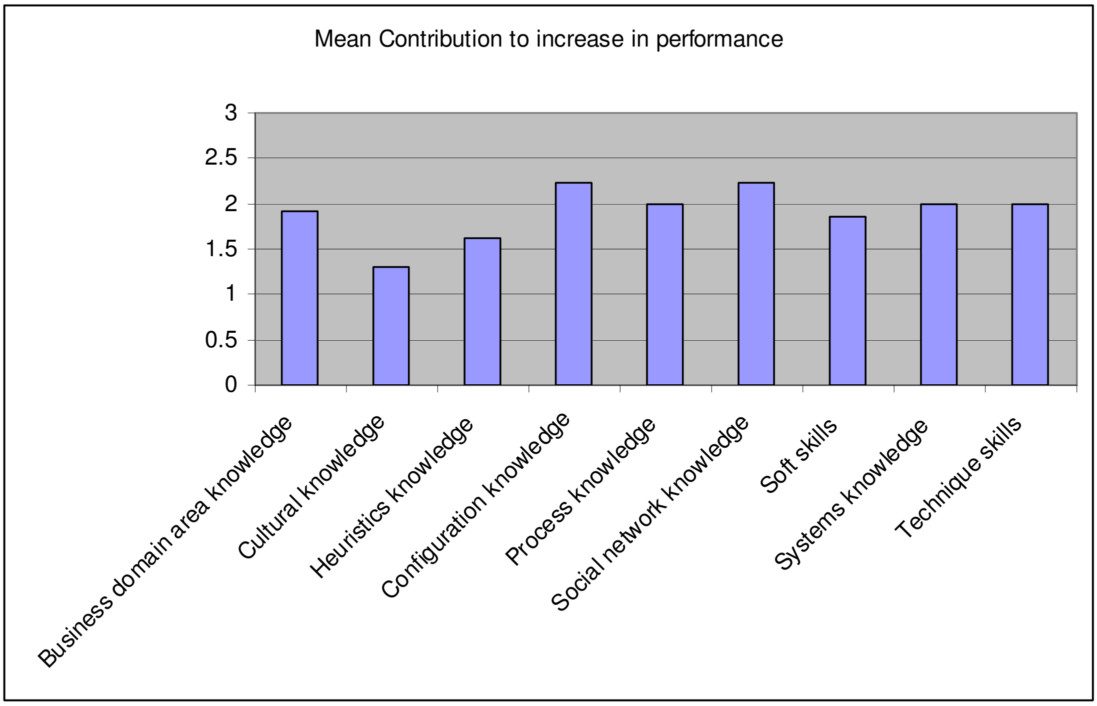

Figure 6.12. Knowledge factors : Mean contribution to increase in performance

6.5.2.2 Dispositional reasons

Dispositions are individual's tendencies to put their capabilities into action (Perkins et al., 1993). Rasch and Tosi (1992) demonstrated that achievement needs, effort and self-esteem and personal reasons contribute to performance in software development albeit at a lower level than ability. During the formulation of theoretical foundation for the research (see section 3.2.1.6), it was argued that dispositions affect the way individuals act (Perkins et al., 1993). Consistent with the literature (see section 4 2.3.2) the case study results show support to the propositions that achievement needs, effort and self-esteem contribute to increase in performance (see Table 6.7). It is found that effort and achievement needs contribute more to improvements in performance than increase in cultural and heuristics knowledge (see ranking in Table 6.7).

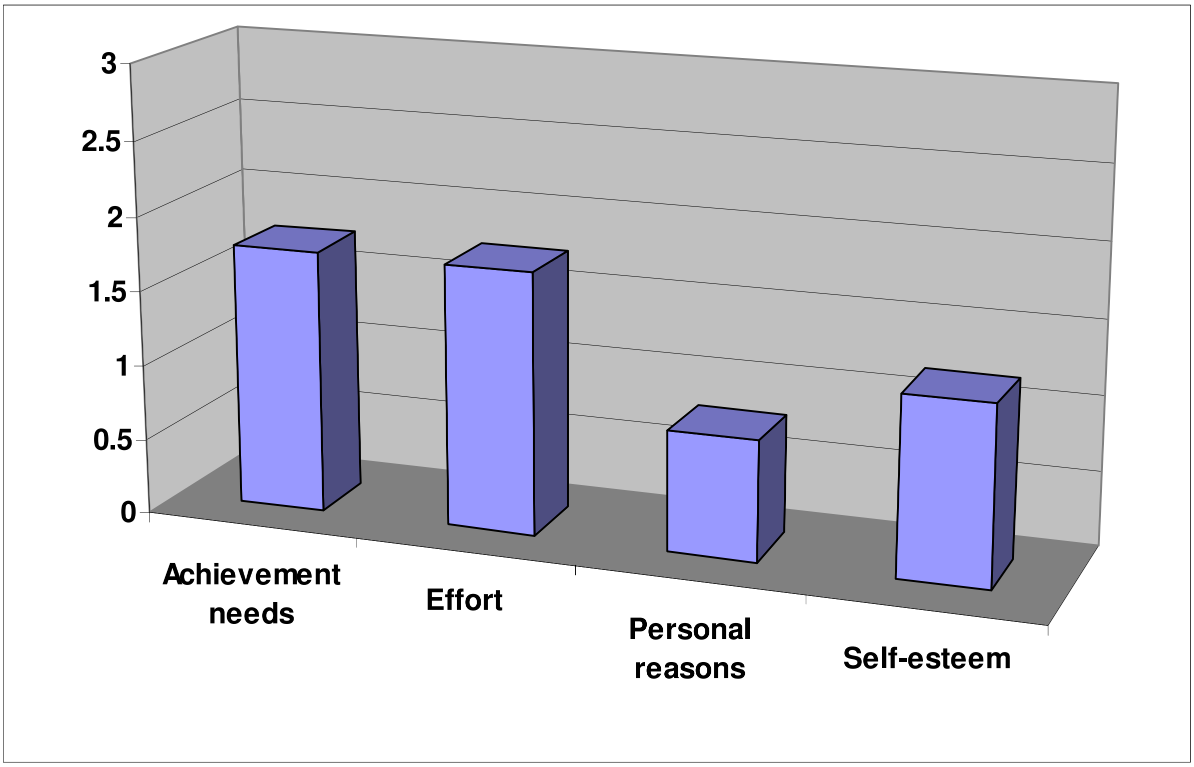

Figure 6.13. Contribution of dispositional reasons to increased performance

Thus, the results of the case study provide empirical support to the proposition that dispositions such as achievement needs, effort, self-esteem and personal reasons play a part in the performance of an individual.

6.5.2.3 Analysis based on sub-categorisation

During review of the case study data, some noticeable patterns could be seen based on categorisation of the participants. Further analysis revealed some interesting findings. The participants were categorised as follows, and data was analysed:

- Newcomers to the organisation – At the time the individual joined the SE team, he/she was new to the organisation. The result of this analysis is presented in Table 6.8.
- Experienced in organisation, but newcomer to the project team – At the time the individual joined the SE team, they had worked in different parts of the organisation. The result of this analysis is presented in Table 6.9.

- Experienced in the application system as well as organisation – At the time the individual joined the SE team, he/she had prior experience with the team as well as the organisation. Such individuals came back to perform the job in the project team after a stint elsewhere. The result of this analysis is presented in Table 6.10.

Newcomers to the organisation

The data from the case study team shows us that the newcomer to the organisation have a low level of knowledge in the case of the following types of knowledge (see Figure 6.14).

- Business domain area knowledge
- Cultural knowledge
- Application software items (Configuration knowledge)
- Social network knowledge
- Heuristics

All the above knowledge types (except business domain area knowledge) have been identified to be organisational contextual knowledge types[40] (see section 3.3.3.5). Business domain area knowledge can also be thought of as a contextual knowledge for an individual if the individual has not worked in similar business domain before. Hence, as can be expected, it is found that newcomers joining an organisation will not posses organisational specific knowledge (organisational contextual knowledge).

Newcomers have a low level of contextual knowledge. As the newcomers start to work on the project team, they gain the contextual knowledge required. Figure 6.15 shows that both contextual knowledge types and non-contextual knowledge types have increased for the newcomer after working in the project. This is further evidence to reconciled social and cognitive origin theories that individuals learn new contextual and cognitive knowledge after participating in a goal directed activity (Billett, 1995; Billett, 1996; Sveiby, 1997; Shariq, 1999; Packer and Goicoechea, 2000; Augier et al., 2001; Mason, 2007). Thus, based on this single embedded case study, it is observed that newcomers to the organisation have a low level of the contextual knowledge required to perform their job.

40 Process knowledge was also identified as a predominantly contextual knowledge type. However, it is found that newcomers posses some level of process knowledge.

Knowledge factor	Increase in Knowledge index (%)	Contribution to increase in performance (%)	Initial knowledge index	Current knowledge index
Business domain area	33.33	2.00	0.00	33.33
Cultural	41.67	2.00	0.00	41.67
Heuristics	16.67	1.00	8.33	25.00
knowledge of application software Items*	4.24	2.00	0.00	4.24
Processes and procedures	0.00	2.00	33.33	33.33
Social network knowledge	12.50	2.00	0.00	12.50
Soft skills	0.00	1.00	85.19	85.19
Systems knowledge	0.00	2.00	100.00	100.00
Technique skills	5.13	1.00	25.64	30.77

Table 6.8. Summary of averages for newcomer to project team

Knowledge factor	Increase in Knowledge index (%)	Contribution to increase in performance (%)	Initial knowledge index	Current knowledge index
Business domain area	0.00	2.00	74.07	74.07
Cultural	12.73	1.44	46.99	59.72
Heuristics	14.81	1.67	46.30	61.11
knowledge of application software Items	35.42	2.56	6.34	41.75
Processes and procedures	40.74	2.00	29.63	70.37
Social network knowledge	43.33	2.44	17.04	60.37
Soft skills	1.48	1.89	67.84	69.32
Systems knowledge	5.56	2.33	59.26	64.81
Technique skills	8.83	2.11	35.61	44.44

Table 6.9. Mean: Individuals - experienced in org – newcomer to project team

Knowledge factor	Increase in Knowledge index (%)	Contribution to increase in performance (%)	Initial knowledge index	Current knowledge index
Business domain area	11.11	1.67	55.55	66.66
Cultural	13.19	0.67	42.36	55.55
Heuristics	11.11	1.67	55.55	66.66
Application software items	1.65	1.33	38.27	39.92
Process and procedures	11.11	2.00	66.66	77.78
Social network	13.33	1.67	42.22	55.56
Soft skills	0.65	2.00	75.16	75.81
Systems knowledge	5.56	1.00	83.33	88.89
Technique skills	3.42	2.00	60.68	64.10

Table 6.10. Mean: Individuals - experienced in systems as well as organisation

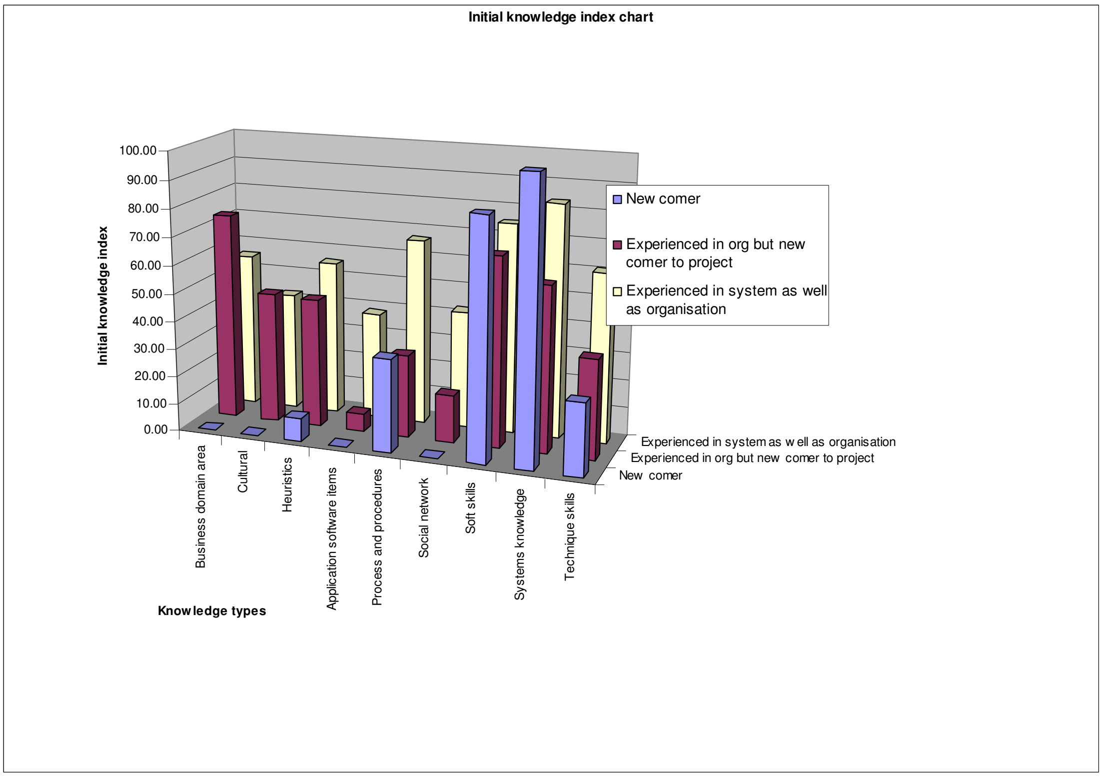

Figure 6.14. **Initial knowledge levels**

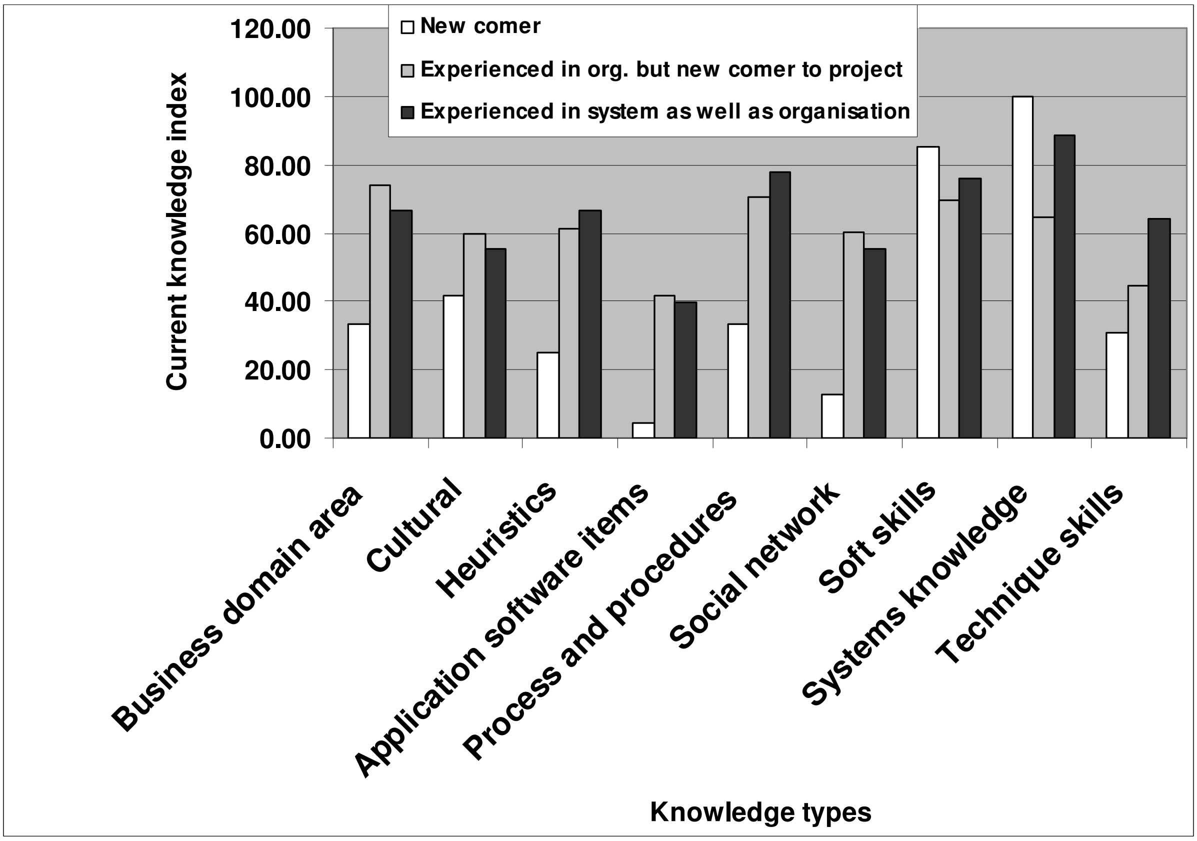

Figure 6.15. **Current knowledge levels**

Users possess configuration knowledge and business domain area knowledge

In the case study team, two business analyst/testers came from a user background. These users were seconded from Operations division to the case study team and reported to the team project manager. The qualitative data collected from two business-analyst-testers are summarised here below:

BA-Tester-1 said that her background as an operational user of the CMS application system helped her to perform the job effectively as a business analyst-tester.

BA-Tester-2 had an excellent knowledge of the application system and its functionality even though her knowledge of batch processes is limited. She had full knowledge of the correspondence issued to clients, user interfaces and the functionality of the system. BA-Tester-2 believed that she was brought into the team because of this knowledge of the application system functionality.

These two cases suggest that users of the system possess configuration knowledge and business domain area knowledge. An analysis of configuration knowledge levels for these users turned business analyst/testers shows that they possessed configuration knowledge when they joined the team. BA-Tester-2 reported an initial configuration knowledge index[41] of 23.62% and BA-Tester-1 reported an initial configuration knowledge index of 13.33%. Both users report an initial business domain area knowledge index of 66.66%. This shows that they came into the team with reasonable levels of configuration knowledge and business domain area knowledge. Earlier, it was found that newcomers possess a low level of configuration knowledge (see Figure 6.14). Getting new people with configuration knowledge is difficult since newcomers won't posses them. But, this case study research provides empirical support that the users of the system possess configuration knowledge. This finding provides support to existing literature that embedding users in SE teams (Butler and Fitzgerald, 2001), will help in alleviating the problems of thin spread of knowledge and conflicting and fluctuating requirements (Curtis et al., 1988), reduce learning times (Walz et al., 1993; He, 2004) and will lead to SE success .

[41] Normalised percentage index

In conclusion, it is found from the case study that users working as analysts/testers bring in a wealth of business domain area knowledge and application system knowledge (configuration knowledge). This can alleviate problems faced by teams such as conflicting and fluctuating requirements (Curtis et al., 1988) and can lead to success. Based on these findings the researcher recommends that users of the system should be provided with appropriate skills on business analysis and testing and should be engaged as business analyst/testers to make use of their configuration knowledge.

Members who have worked in the organisation before joining the team

Members of the case study team who have worked in the organisation but not in the case study team, reported some level of initial cultural knowledge, configuration knowledge and social network knowledge (see Figure 6.14). Such members have worked in the organisation before and as part of performing their duties appear to have gained the contextual knowledge. Hence, this finding shows support to reconciled social and cognitive origin theories (see section 3.4) (Billett, 1995; Billett, 1996; Sveiby, 1997; Shariq, 1999; Packer and Goicoechea, 2000; Augier et al., 2001; Mason, 2007) where individuals learn new knowledge after being involved in goal directed activity. Note that this observation is in contrast to the other finding that newcomers to the organisation have a low level of the contextual knowledge.

Members who have worked in the organisation and on the application system report contextual knowledge

As can be expected, team members, who have worked in the organisation and on the application system, reported some level of cultural knowledge, configuration knowledge and social network knowledge apart from other types of knowledge (See Figure 6.14). This observation suggests that members, who have worked in the SE team, hold valuable contextual knowledge.

6.5.3 Findings and conclusions

The case study research conducted in this study empirically provides support to the theoretical framework proposed in this research in that all knowledge types contribute[42] to improved performance. In the case study team, based on Figure 6.12, it was found that knowledge types identified in the theoretical framework have been found to contribute to increase in performance. In the case study team, it was found that the configuration knowledge and social

[42] through averages only; not statistical relationship.

network knowledge rank as the top contributors to improvements in performance (see ranking in Figure 6.12).

It was found that identified knowledge factors and dispositional factors contribute to increase in performance. These findings are important for both theory and practice. It is advanced that all knowledge factors contribute to performance. From a practice point of view, targeted KM activities can be conducted to increase higher ranked knowledge types. For instance, focussed efforts to identify and impart configuration knowledge to new comers are likely to be of assistance. Similarly, addressing the codification of social network knowledge during transition is likely to be beneficial to performance. Thus, these findings from this research are potentially very useful better KM in SE in reducing failures (see section 1.4 for more on how the results can be applied). Further, SE academic curriculum can be designed to include the identified knowledge types and emphasise their importance to increased performance.

Section 6.5.2.3 presented the analysis based on sub-categorisations and the resulting findings are summarised below. These findings are potentially useful in recruitment and training of individuals for SE teams. Some of the implications of these findings to KM strategies are presented below:

- New comers have a low level of knowledge in contextual knowledge types such as business domain area knowledge, cultural knowledge, configuration knowledge, social network knowledge and heuristics knowledge types. Hence, strategies, such as training, mentoring and guidance, should be put in place to impart these knowledge types to newcomers.
- Users of the system possess necessary configuration knowledge. Hence, embedding users into SE teams so that the configuration knowledge levels are increased within teams.
- Existing members, who have worked in the organisation before but not on the application system/project team, possess some level of cultural knowledge and social network knowledge. Hence, recruiting individuals with such knowledge can help in smoother transition of the team particularly if there are many newcomers to the teams.
- Existing members, who have both worked on the application system as well as in the organisation, report some levels of cultural, social network and configuration

knowledge. Hence, strategies should be developed to retain existing members holding the knowledge and transitioning that knowledge to other members of the team.

6.6 Limitations of this research project

6.6.1 Issues in generalisation

This chapter presented a single embedded, predominantly quantitative, case study. Case study research cannot generalise results to a population. Case studies cannot claim statistical generalisation as a method of generalising the results of case study (Lee and Baskerville, 2003; Yin, 2003). The results of the case study can be extended to theoretic proportions only if multiple case studies are conducted and patterns revealed. With one embedded case study, albeit with multiple units, no generalisation is claimed. However, the results of the case study research provide an insight into knowledge and skill requirements and their contribution to performance in a real life context. Thus, the findings from the case study are a useful contribution to both theory and to practice by classifying and identifying those knowledge elements essential to SE roles and knowledge contribution to performance.

6.6.2 Other limitations

Environmental factors can affect the results of a case study. These are discussed in this sub-section.

6.6.2.1 Business analyst/tester role

Many organisations have business analyst role and tester role separated and performed by different individuals. In this case study team one individual performs a joint role. Hence, the results of the study must be applied carefully to other organisations where the roles are separated out.

6.6.2.2 Measuring knowledge

Knowledge is an in-tangible phenomenon that cannot be easily measured (Freeman, 2001). It is in the mind of the individuals (Davenport and Prusak, 1998). This research collects the perceptions of individuals relating to knowledge requirements and contribution of knowledge to performance. It is not clear how the perceived knowledge requirement would be different from the actual knowledge requirements for performing job. Inarticulable tacit knowledge (Busch et al., 2001) may not be captured as part of this research.

The project-survey questionnaire addresses the following constructs:

- Perceived knowledge at the time you joined the project team
- Perceived current knowledge

- Perceived extent to which the knowledge item helped you to perform your job

One criticism could be that people could easily forget or may mistakenly misrepresent past knowledge. A longitudinal study may have alleviated this limitation but time consuming. This is a limitation of the study. This is a limitation that we need to consider while applying findings of this research.

6.7 Reflections on the study

The case study research was an interesting and fulfilling experience for the researcher. He fulfilled a dual role in the process, as a researcher and as a consultant in the project team. The following are some reflections and thoughts from the researcher as he looks back at the plan, data collection and analysis phases.

6.7.1 Advantages of being an insider

The researcher had access to all the members of the team and had known them before. Establishing the contact and building the rapport with them was much easier. The researcher possessed the knowledge of the application software system. Being an insider into the case study team helped in planning and conducting the research.

6.7.2 Disadvantages of being an insider

Issues of bias (see 4.5.6) are a key disadvantage of being an insider. In order to manage the issue of bias, the researcher had to be very careful in keeping his opinions and information to himself and look at the facts objectively. Hence, the researcher designed a more positivistic and quantitative field study type project.

6.8 Chapter Summary

SE is a knowledge intensive activity. If we don’t know the different types of knowledge required to perform various roles in SE, then we cannot effectively manage them. The research set out to answer two research questions relating to identifying this knowledge and how they contribute to increases in performance. An embedded case study research was conducted in a large government organisation. Thirteen members participated in the study from a project team, which included business analyst/testers, programmer-analysts, project managers and a test and implementation manager. Both qualitative and quantitative data were collected on the knowledge requirements and contributions to improved performance for each role.

The case study research provides empirical evidence to the theoretical framework proposed in this research. It was found that knowledge types identified in the theoretical framework were important in performing various roles in the case study SE team. It was observed that the project manager required more social network knowledge and cultural knowledge than configuration knowledge. The programmer-analyst had a lower need for cultural knowledge than other roles analysed in this study.

No generalisation is claimed from the case study results. However, the results of this study provide an insight into SE knowledge requirements and knowledge factors contributing to performance improvements. Hence, the results of the study could be helpful in formulating training plans, recruitment, job rotation planning and, most importantly, knowledge and performance management.

7 Integration and further exploration

7.1 Introduction

The aim of this research is to identify knowledge requirements and factors contributing to improved performance in SE teams. A theoretical framework between knowledge factors and performance has been developed as the basis for this research in order to answer the research questions. This chapter presents the reconciliation of the findings from case study research (Chapter 6) and general-survey research (Chapter 5). This chapter presents the EFA of knowledge requirements data to understand hidden structures and patterns.

The organisation of this chapter is presented in Figure 7.1. Following this introduction, section 7.2 presents the reconciliation between case study research methods and survey research methods. Reclassification using Exploratory Factor Analysis (EFA) of collected data is presented in section 7.3. Analysis was carried out to look for the hidden structure of factors within the data using factor analysis. This section also presents the transformation of the identified factors in theoretical framework into the new structure and how the new factors support the underlying theories. This section, further, looks at any relationships that exist between demographic profile items and the identified factors.

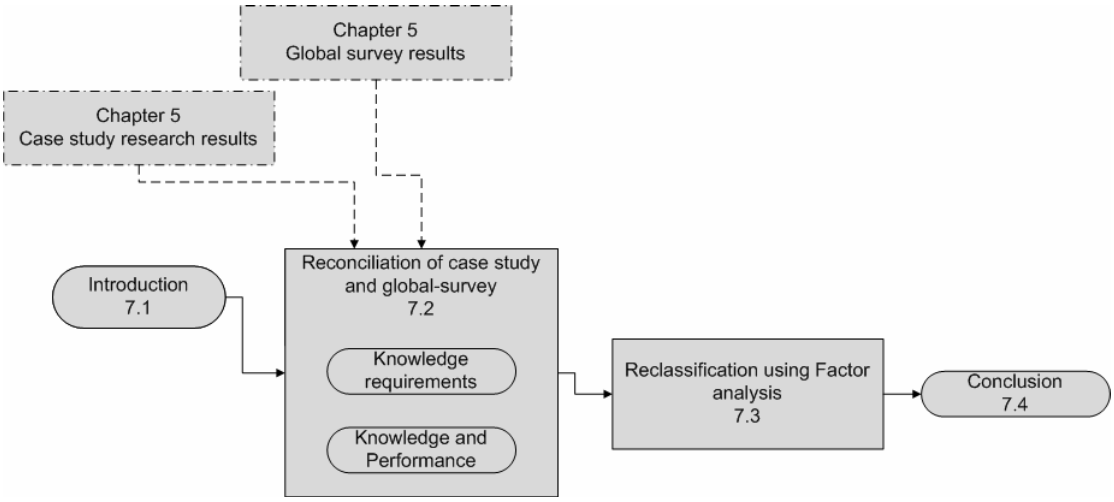

Figure 7.1. Flowchart: Organisation of Chapter 7

7.2 Reconciliation of case study and general-survey findings

7.2.1 Introduction

Two research studies were used to find answers to the posed research questions. The thesis has reported findings from a case study of a real-life SE team (Chapter 6) and a general-survey of SE professionals (Chapter 5). As will be seen in this chapter, the findings from these studies indicate that the empirical evidence (both quantitative and qualitative) collected are aligned between the two research methods to provide support to the theoretical framework and to answer the research questions. A number of additional findings could be made from each of these research studies independently.

One of the key differences between the conducted case study research and the general-survey research is that the case study research considered less number of roles. For example, the key roles considered in the case study consisted of programmer-analyst, business analyst-tester and project manager whereas, the general-survey was able to collect responses from business analyst, programmers, designers, architects and programmer-analysts. This difference needs to be taken into account when reconciling the results.

The reconciliation of findings from the case study and general-survey on knowledge requirements to perform various roles in SE is presented in sub-section 7.2.2. The second research question on knowledge factors that contribute to improved performance was assessed by using both case study and general-survey research results. The reconciliation of findings for this research question is presented in sub-section 7.2.3.

7.2.2 Knowledge requirements to perform various roles

As we will see in this section, the results of the case study research (see section 6.4 for findings) as well as the general-survey (see section 5.3 for findings), substantiate each other and provide support to the proposition that knowledge and skill requirements identified as part of the theoretical framework are required to perform various roles in SE teams.

Both the general-survey and case study provide evidence that cultural knowledge is important for all roles. In the case study team project managers required cultural knowledge at the highest level while programmers and programmer-analysts required a lower level of cultural knowledge than other roles.

The findings from both studies show that configuration knowledge is required to perform all roles in SE. The case study research was able to delve into details on configuration knowledge requirements for various roles in the team. Based on this analysis of configuration knowledge requirements the following findings are made: Programmers and programmer-analysts require configuration knowledge at a comparatively higher level than Project Managers. Business analysts, as part of their role in developing specifications, require some of this knowledge such as user interfaces, functionality and external interfaces at a higher level than other parts such as reusable functionality. In contrast, project managers only need to have a broad understanding of the application software system. Programmer-analysts and business analyst/tester require in depth knowledge of some of the software items and a broad understanding of others.

Process knowledge is necessary to perform various SE roles. It is found from both general-survey as well as case study that all SE roles require soft skills to perform their job. Additionally, the findings from the case study shows that project managers and business-analysts require higher levels of soft skills in comparison to programmers and programmer-analysts.

The results from both studies show that the technique skills are important to perform various roles in SE. In addition, the case study findings show that different roles require different kinds of technique skills. Business analysts require business analysis and testing skills. Programmers require technical skills while programmer-analysts also need to know application software architectures.

The results from the general-survey and case study show that social network knowledge is important to perform various SE roles. Project managers require higher levels of social network knowledge and are followed by business-analysts and programmer-analysts. Programmers require the least social network knowledge to perform their job. Both the general-survey and case study show that heuristics knowledge is important to perform various roles in SE. Both the general-survey and case study research results show that all roles require systems knowledge and business domain knowledge. Case study results show that project managers require more of business domain knowledge than other roles.

7.2.3 Knowledge factors to improved performance

Both the case study and general-survey results expected that all knowledge factors in the theoretical framework contribute to increase in performance. From both studies, the relative contribution of different types of knowledge to increase in performance is ranked. This comparison is shown in Table 7.1.

No.	Type of knowledge	Finalised ranking from general-survey	Ranking from case study#
1.	Technique skills	1	3
2.	Soft skills	5	7
3.	Business domain knowledge	4	6
4.	Process knowledge	7	3
5.	Systems knowledge	3	3
6.	Social network knowledge	6	1
7.	Cultural knowledge	9	8
8.	Heuristics knowledge	8	7
9.	Configuration knowledge	2	1

Table 7.1. Comparison of rankings from case study and general-survey

Dispositional factors excluded. Only knowledge factors included.

The results from both studies show that configuration knowledge, systems knowledge and technique skills show higher rankings in terms of their contribution to improvements in performance. On the other hand, soft skills, heuristics knowledge and cultural knowledge show a low contribution to improvements in performance.

There have been differences in rankings for process knowledge and social network knowledge. While, case study results show that social network knowledge is ranked number one in its contribution to improvement in performance, it ranks 6^{th} in global-survey findings. Similarly, process knowledge, which ranks 3^{rd} in case study findings, is ranked 7^{th} in general-survey findings. The case study has captured the perceptions of real life participants in a specific organisational context. The contextual and environmental factors in a specific case study team within an organisation can lead to varied rankings of contribution of different knowledge types to performance. Hence, such variation can be expected between a specific context and the overall summary of a sample of a population. The aim of the case study, for this research, was to capture real-world context. Hence, this real world study could deviate from the average in many respects.

It is important to note that both general-survey and case study have found that all the factors identified in the theoretical framework have an implication for improved performance. Survey research, due to its nature, collects large quantitative data from samples of the population. Hence, the general-survey research results are taken to be the finding of this research.

It was found from case study that dispositional factors such as achievement needs, effort and self-esteem and personal reasons are also important contributors to increase in performance. This finding substantiates existing literature (Rasch and Tosi, 1992).

Apart from the above conclusions, the following findings are made from the case study research:

- New comers to the organisation have a low level of the following contextual knowledge types:
    - Cultural knowledge
    - Configuration knowledge
    - Social network knowledge
    - Heuristics knowledge
    - Business domain area knowledge
- Users of the system possess necessary configuration knowledge and they are a good source for recruitment.
- Existing members, who have worked in the organisation before but not on the application system/project team, possess some level of cultural knowledge and social network knowledge.
- Existing members, who have both worked on the application system as well as in the organisation, report some levels of cultural, social network and configuration knowledge.

In addition, the general-survey research found a number of relationships between demographic profile items and perceived increase in knowledge and perceived contribution of increase in knowledge to increase in performance. These are summarised below.

- It is found that as the Total Professional Experience increases, the perceived contribution of process knowledge, general systems knowledge and cultural knowledge to improvements in performance declines.
- As the Experience within SE team increases there is higher perceived contribution of increase in technique skills and configuration knowledge to increase in performance.

- There is a statistically significant relation between role and perceived contribution of increase in configuration knowledge to increase in performance. The contribution is higher for programmers, programmer-analyst, designers and architects. It is comparatively lower for project managers, testers and business analysts.

These findings have substantial relevance to theory and practice. From a theoretical point of view, new ground is covered by identifying the knowledge factors and their contribution to performance. It is found from case study that contextual knowledge types are not held by newcomers to the organisation but are held by individuals who have worked in the organisation. The relations between different types of experience and knowledge types contribution to performance is substantive contribution to theory. Apart from contributions to theory, the results of the study have application to practice. Focussed KM activities can be carried out to improve knowledge of individuals through strategies such as training, knowledge continuity management, mentoring. This will improve performance and success rates. Further, academic curriculum can be designed keeping these knowledge types in mind.

7.2.4 Summary

This section reconciled the results from the case study and from the general-survey findings. It is found that both the case study research results and the general-survey research results are in alignment. First, it is found from the case study and the general-survey that all knowledge factors identified in the theoretical framework are required to perform various roles. It is found that the increase of the knowledge factors identified in the theoretical framework contribute to improved performance. However, there are differences between case study and general-survey findings in the rankings of the level of contribution to improved performance with respect to social network knowledge and process knowledge. Such a result can be expected since case study is influenced by the organisational and contextual circumstances. Additionally, the case study findings show that users of the system possess configuration knowledge.

7.3 Knowledge requirements: Re-classification using Exploratory Factor Analysis (EFA)

7.3.1 Introduction

During general-survey analysis, it was found that some of the original factors, derived from the literature, showed borderline on reliability (see 5.3.4.1). This suggests scope for re-classification using methods such as factor analysis. An exploratory factor analysis of data can reveal underlying patterns within the data. Hence, EFA was carried out on the data. This section reports the results of EFA. The results of factor analysis are then used to analyse further relationships between demographic profile and knowledge factors.

7.3.2 EFA Findings: Reclassified Framework

The general-survey research has a sample size of 178 and 50 variables in the questionnaire (item to sample ratio of 1: 3.56). Guadagnoli and Velicer (1988, cited in Stanek, 1993) suggest 100 to 200 observations as enough. Garson (1998) suggests a ration of 1: 10. Costello and Osborne (2005) find that large percentage of researchers report factor analyses using relatively low samples (item to sample ratio of 1: 2) and there is no definitive scientific guide to sample size requirement (Costello and Osborne, 2005; Garson, 2007). The sample size of 178 with a ratio of 1: 3.56, while being low is common practice and is adequate for analysis. Values of 0.866 for KMO and 0.000 for Bartlett's test of sphericity significance suggests the data is meritorious for factorability (Hair et al., 1998; Yaobin and Tao, 2007).

Principal Axis Factoring (PAF), a widely used method for EFA (Child, 1990; Costello and Osborne, 2005), is used for factor extraction. As suggested by Garson (1998), various combinations of options (rotations, cut-off) and interpretations were trialled. Promax rotation method was finalised since it produced the best interpretable factor structure. The number of factors in the theoretical framework proposed in this research is 9. As suggested by Costello and Osborne (2005), the researcher ran EFA in SPSS with various numbers of factors (5, 6, 7,8,9, 10 and 11) and the resulting rotated item loading tables were compared. With number of factors more than 10, there was weak loading and hence was rejected (Costello and Osborne, 2005). With number of factors as 5, some of the items got crowded and loading values suffered for some items. Number of factors as 7 was best interpretable and meaningful (Rummel, 1970), with strong loading of items and hence was chosen.

The Table 7.2 and Table 7.3 show the rotated factor matrix and associated statistical values. The choice of cut-off value[43] for loading is a matter of the research preference and interpretability of the factors (Tabachnick and Fidell, 1989). A cut off of 0.4 is fair and provided the best interpretable results and hence was chosen. The process of interpretation is coloured by the researcher's perspective (Rummel, 1970) and is an art as well as science (Tabachnick and Fidell, 1989). Interpretation of results are carried out considering that the factors as underlying classifications. A descriptive approach is used to name the factors, with names communicating their substantive nature to others (Rummel, 1970). Appendix section Error! Reference source not found. shows the item correlation matrix and section 11.11 shows the factor correlation matrix.

43 Comrey (1973 cited in Tabachnick and Fidell, 1989) says that loadings in excess of 0.71 are considered excellent, 0.63 very good, .55 good, .45 fair and 0.32 poor.

Knowledge / Skill Items	Factor						
	1	2	3	4	5	6	7
	Org. Culture and relationships KSA	Project Software dev. KSA	Management skills	Heuristic knowledge	Communication and relationship skills	Business functional KSA	Problem solving abilities
B36 Understanding of the degree to which work activities are organised around teams rather than individuals	**.775**	.136	.378	.408	.368	.354	.179
B35 Understanding of the degree to which management decisions take into consideration the effect of outcomes on people	**.761**	.128	.399	.442	.353	.420	.198
B33 Understanding of the expectations of organisation with respect to attention to detail	**.738**	.339	.470	.324	.241	.339	.176
B34 Understanding of the degree to which the management focuses on outcomes	**.729**	.246	.321	.354	.315	.289	.239
B32 Understanding of innovation and risk taking nature of organisation	**.702**	.323	.449	.267	.244	.372	.005
B38 Understanding of the aggressiveness of people in organisation	**.697**	.249	.490	.523	.300	.303	.140
B37 Understanding of the knowledge sharing culture within organisation	**.655**	.157	.362	.519	.346	.297	.173
B31 Professional network of people outside the organisation	**.622**	.348	.489	.258	.286	.296	.115
B39 Understanding of the degree to which organisational activities emphasise maintaining the status quo in contrast to growth	**.573**	.122	.366	.489	.277	.215	.207
B27 social network within organisation but outside software project team	**.513**	.217	.431	.223	.218	.400	.134
B30 knowledge of development methodologies used to build the app	**.489**	.466	.440	.389	.254	.420	.119
B49 knowledge of the overall architecture of the application	.219	**.790**	.150	.238	.232	.265	.245
B50 knowledge of the software development environment at the software project team	.309	**.745**	.218	.321	.251	.388	.206
B25 knowledge of application software architectures	.196	**.703**	.166	.136	.140	.236	.171
B48 knowledge of database structures of the application system	.198	**.670**	.116	.238	.114	.461	.173
B18 Software skills required to develop the application software	.033	**.620**	.033	.159	.107	.306	.171
B47 knowledge of any reusable component functionality within the	.354	**.583**	.200	.290	.242	.452	.119
B24 knowledge of software development processes	.382	**.518**	.488	.374	.293	.408	.042
B3 Adaptability to new technologies languages	.181	**.411**	.245	.210	.232	.160	.089
B19 Hardware skills required to develop the application software	.074	.332	.138	.155	-.076	.058	.065
B15 Idea initiation skills	.503	.312	**.667**	.462	.280	.283	.176
B10 Leadership skills	.411	.073	**.651**	.345	.451	.313	.174
B11 Negotiation skills	.317	.010	**.641**	.243	.329	.229	.215
B14 General writing skills	.463	.128	**.622**	.474	.445	.468	.154
B7 Organisational skills	.420	.147	**.619**	.324	.472	.242	.287
B8 Ability to give and receive constructive criticism	.375	.146	**.607**	.354	.274	.227	.098
B16 Listening skills	.488	.197	**.570**	.403	.562	.359	.258
B22 Project management skills	.442	.140	**.503**	.247	.362	.281	.045
B9 Ability to multitask	.162	.201	**.503**	.340	.365	.137	.331
B17 Technical writing skills	.383	.206	**.480**	.306	.250	.404	.019
B13 Stress management skills	.424	.300	**.468**	.405	.341	.323	.309
B41 knowledge of guidelines within the organisation	.520	.255	.474	**.818**	.386	.389	.257
B42 knowledge of standards within the organisation	.455	.265	.410	**.797**	.374	.330	.121
B43 knowledge of rules of thumb to perform various jobs	.417	.387	.412	**.752**	.381	.442	.181
B40 knowledge of shortcuts to accomplish various tasks	.279	.249	.378	**.643**	.070	.322	.315
B5 inter team communication	.276	.274	.428	.302	**.779**	.289	.174
B4 Verbal communication skills	.290	.249	.522	.284	**.710**	.367	.222
B12 interpersonal skills	.307	.188	.460	.361	**.625**	.319	.495
B2 Time management skills	.365	.122	.422	.238	**.585**	.230	.230
B1 team work skills long term	.337	.074	.240	.252	**.535**	.081	.325
B44 knowledge of user interfaces within the application system	.413	.464	.363	.485	.325	**.778**	.125
B46 knowledge of program functionality within the system	.283	.604	.187	.278	.245	**.694**	.085
B45 knowledge of the interfaces of the application system with external systems	.433	.546	.334	.454	.310	**.686**	.070
B23 Business domain area knowledge relating to the applications	.381	.270	.417	.256	.287	**.480**	.218
B20 Business analysis skills	.280	.065	.390	.156	.331	**.466**	.369
B21 Testing skills	.166	.209	.202	.220	.117	**.453**	.304
B28 General problem solving skills	.257	.355	.214	.277	.393	.246	**.595**
B26 Understanding of who knows what within software project	.380	.405	.239	.375	.337	.337	**.548**
B29 Analytical and modelling skills	.328	.249	.332	.275	.203	.296	**.524**
B6 investigative skills	.118	.308	.483	.187	.454	.162	**.518**

Table 7.2. Rotated factor matrix

Factors were extracted using Principal Axis Factoring and Promax rotation with Kaiser Normalisation. Rotation converged in 11 iterations.

Knowledge / Skill Items	Factor						
	1	2	3	4	5	6	7
	Organisational Culture and relationships KSA	Project Software development KSA	Management skills	Heuristic knowledge	Communication and relationship skills	Business functional KSA	Problem solving abilities
Initial Eigen values	13.804	4.033	2.621	1.872	1.759	1.652	1.448
Initial Eigen values - % of Variance	27.607	8.065	5.242	3.744	3.518	3.305	2.896
Extraction sums of squared loadings – Total	13.302	3.538	2.136	1.430	1.209	1.100	0.925
Extraction sums of squared loadings - % of Variance	26.603	7.075	4.272	2.860	2.418	2.199	1.850
Rotation Total	9.503	6.576	8.816	7.399	6.519	6.761	3.210
Reliability (Cronbach's alpha)	0.9	0.839	0.872	0.847	0.781	0.78	0.719

Table 7.3. Factor analysis: statistical results.

7.3.2.2 Factor 1: Organisational culture and relationships KSA

Factor 1 is a combination of organisational cultural items together with the social network outside the software project team (both within and outside the organisation) items. This suggests that the knowledge of culture of the overall organisation and organisational social network knowledge are related. Busch et al. (2001) carried out social network analysis of an systems organisation and demonstrated the pivotal role of individuals in tacit knowledge transmission. They argued that tacit knowledge, at the individual level, when congregated into organisational level becomes culture. The discovery of *Factor 1*, which shows the association between organisational cultural knowledge and organisational social network knowledge, provides support to this finding. Both organisational culture and relationships within the organisation are largely contextual to the organisation (see section 3.3.3).

Note that the *social network knowledge within the team* did not load onto this factor. If the correlation between tacit knowledge flows and culture is true, it would be expected that *social network knowledge within the team* should also have been loaded onto this factor. While this item loaded with a lower score of 0.380 in Factor 1, it is found that *social network knowledge within the team* has a higher association with Factor 7: Problem solving skills than organisational culture. As we will see later (see sub-section 7.3.2.8), this suggests that individuals consult extensively with others in their team to solve problems.

The item *knowledge of development methodologies used to build the application* loaded onto this factor with a load of 0.489. This same item also loaded into Factor 2 at a load of 0.466. Development methodologies are standard notations and embodiments of good practice (Sommerville, 2001) and hence can be termed prior cognitive skills that are not circumstantial. On the other hand methodologies adopted in organisations are quite different from original prescriptions (Mathiassen and Vogelsang, 2005a) leading to a unique and local method in practice (Bansler and Bodker, 1993; Sommerville, 2001; Madsen et al., 2006). Hence, the knowledge of development methodologies includes contextual and non-contextual parts. The loading of this item into Factor 1 suggests that knowledge of development methodologies has a higher association with organisational context and is consistent with the literature.

7.3.2.3 Factor 2: Project software development KSA

Some of the configuration knowledge items (such as *knowledge of database structures* and *knowledge of reusable component functionality within the application system*) in the original theoretical framework have grouped together with the items *1) software skills required to perform application development, 2) knowledge of processes, 3) adaptability to new technologies and languages, and 4) knowledge of software development environment*. This factor relates to the expert knowledge, skills and abilities required to perform various application software development functions. Hence, this factor is named as the 'Project software development KSA'.

SE roles require both non-contextual cognitive knowledge and contextual knowledge. It has been identified from literature (see section 3.3.3) that the knowledge types a) processes and procedures and b) the application system knowledge are largely contextual to the organisation. An individual performing SE role needs to possess the cognitive software skills required for application development, which is non-contextual in nature (see section 3.2.2). According to Billett's (1995) model of Knowledge and Performance, both cognitive and social circumstantial knowledge are required to perform an expert role. Thus, the grouping of items into this factor provides empirical evidence to reconciled social and cognitive origin theories of knowledge and action (Billett, 1995; Billett, 1996; Sveiby, 1997; Shariq, 1999; Packer and Goicoechea, 2000; Augier et al., 2001; Mason, 2007).

Interestingly enough, the item *adaptability to new technologies and languages* was considered as a soft skill in the initial theoretical framework. However, the results of factor analysis show its association to software development. The items 1) *knowledge of user interfaces,* 2) *knowledge of program functionality within the system*, 3) *knowledge of the interfaces of the application system with external systems* did not load into this factor. These have loaded at higher levels in Factor 6. It is argued later that these items have higher association with business functionality.

The cut-off loading value of 0.4 filters hardware skills out of this group. This low loading and its subsequent drop-off from all factors shows that hardware skills have a low requirement level for performing various SE roles.

7.3.2.4 Factor 3: Management skills

The Factor 3 grouping of items relates to leadership and management skills. Project management skill, which was considered to be a part of the technique skills required for project managers in the original theoretical framework, has grouped under this factor. Schwalbe (2002) suggests management skills (including communication, negotiation, leadership and political skills) and organisational skills (such as teamwork, coping skills including the ability to plan, analyse, set and achieve project goals, team work skills) for project managers. This factor closely resembles Schwalbe's (2002) suggestions. Due to it's predominantly management oriented nature, this factor is named as Factor 3: Management skills.

Contrary to expectations, both *general writing skills* and *technical writing skills* loaded into this factor instead of Factor 5: Communication and relationship skills. It appears that writing skills have higher connections to management skills than communication and relationships skills.

7.3.2.5 Factor 4: Heuristic knowledge

Factor 4 contains all of the heuristics knowledge items that were present in the original theoretical framework. The literature review shows that heuristic knowledge is mostly tacit, with some being articulable and others inarticulable (Busch et al., 2001). Heuristic knowledge is both cognitive (see section 3.2.2.5) and organisation specific and has been recognised as needed to perform system development roles (Vitalari and Dickson, 1983; Zmud, 1983; Chan, 2003). Thus, factor analysis results demonstrate support for existing literature.

7.3.2.6 Factor 5: Communication and relationship skills

Factor 5 has grouped items reflecting communication and interpersonal skills along with team work skills. Hence, this factor is named Communication and relationship skills. This grouping shows that to perform SE roles, a combination of communication skills and the ability to work in a team is required. Existing literature has sufficiently emphasised the need for communication and relationships skills (Vitalari, 1985; Hunter and Palvia, 1996; Misic, 1996). Becker et al., (1997) found that communication skills were perceived to be significantly more important for managerial employees than for technical employees. Frampton et al. (2005, cited in Frampton et al., 2006) identified communication skills as one of the key capability for IT architects. Thus, the results of EFA are consistent with the literature in showing the importance of communication skills to perform various roles.

7.3.2.7 Factor 6: Business functional KSA

The items *1) 'knowledge of user interfaces', 2) 'knowledge of program functionality' and 3) 'knowledge of interfaces with external systems*' have grouped into this factor. The business domain area knowledge, user interfaces, program functionality and external interfaces are a representation of business functionality and are largely contextual to the organisation (see section 3.3.3). The importance of this Knowledge of business functionality is recognised in literature (Curtis et al., 1988; Herbsleb and Kuwana, 1993; Walz et al., 1993). Further, this factor includes business analysis skills and testing skills that are largely cognitive in nature (see section 3.2.2). Thus, as Billett (1995) advanced, a combination of organisational contextual knowledge associated with non-contextual cognitive skills forms part of Factor 6. This provides further support for theories of cognitive and social origins of knowledge and action (Billett, 1995; Billett, 1996; Sveiby, 1997; Shariq, 1999; Packer and Goicoechea, 2000; Augier et al., 2001; Mason, 2007).

7.3.2.8 Factor 7: Problem solving abilities

The systems knowledge part of the original theoretical framework (items: 1) *general problem solving skills, 2) analytical and modelling skills*) (see 3.5.2.4) proposed in this research have grouped with the *social network knowledge within the team*. Note that the *investigative skills* which were part of soft skills in the original theoretical framework also loaded onto this factor. At first, these results appeared paradoxical. Upon further analysis of the literature and reflection, the researcher found a number of substantiations for this formulation. For example, problem solving skills require negotiation, ability to resolve conflict and democratic-decision making (Kantrowitz, 2005). Collaborative problem solving is required for SE (Curtis et al., 1988; Rosson, 1996; Cubranic et al., 2004; He, 2004; Ye, 2006), where investigative skills are called upon, to solve complex problems relating to application software systems. Practical problem solving is contextualised where people invent effective problem-strategies (Mayer, 1992). Problem solving models depend upon the contingency task structure, characteristics of the environment, and the experiential familiarity of humans with tasks and environment (Sage and Rouse, 1999). Problem solving activities happen most when the individuals with the right knowledge are accessible to fellow team members (Cross and Parker, 2004). Also, it is known that the other team members receive most interactions from fellow team members (Curtis et al., 1988). Hence, problem solving skills such as *1) investigative skills and 2) analytical* *modelling skills* and *3) general problem solving skills* and *social network knowledge within team* are associated with each other. Hence, in the SE context, it appears that problem solving strategy invented (Mayer, 1992) is through interactions within the team. The EFA findings support this association.

According to Billett's (1995) model of Knowledge and Performance, one's prior cognitive knowledge links together with circumstantial knowledge in participating in a goal directed activity. The grouping of items into this factor shows the association of the circumstantial knowledge of *social network knowledge within the team* and non-contextual cognitive problem solving and analysis skills. Thus, the formulation of this factor lends further support to reconciled cognitive and social origin theories of knowledge and action (Billett, 1995; Billett, 1996; Sveiby, 1997; Shariq, 1999; Packer and Goicoechea, 2000; Augier et al., 2001; Mason, 2007).

7.3.2.9 Transformation

Based on EFA the original theoretical framework proposed in this research has been transformed into a re-classified theoretical framework. This transformation is shown in Figure 7.2. The results of factor analysis show that both social contextual knowledge and non-contextual cognitive knowledge types associate together within the re-classified factors, and are relevant for various roles in SE, thus providing empirical support to reconciled cognitive and social origin theories of knowledge and action (Billett, 1995; Billett, 1996; Sveiby, 1997; Shariq, 1999; Packer and Goicoechea, 2000; Augier et al., 2001; Mason, 2007).

In summary, EFA revealed seven factors relevant to perform SE roles.

- Factor 1 : Organisational culture and relationships KSA (Including organisational culture and organisational relationships)
- Factor 2 : Project software development KSA (Including software skills, knowledge of application software items and processes used in the software project team)
- Factor 3 : Management skills (Including negotiation, Management and organisational skills)
- Factor 4 : Heuristic knowledge (including standards, guidelines, shortcuts and rules of thumb)

- Factor 5: Communication and relationship skills
- Factor 6: Business functional KSA (Including business domain area knowledge, user interfaces, external interfaces, functionality of application, business analysis and testing)
- Factor 7 : Problem solving abilities (Working with others in the team, the ability to investigate, analyse, propose and solve problems and issues)

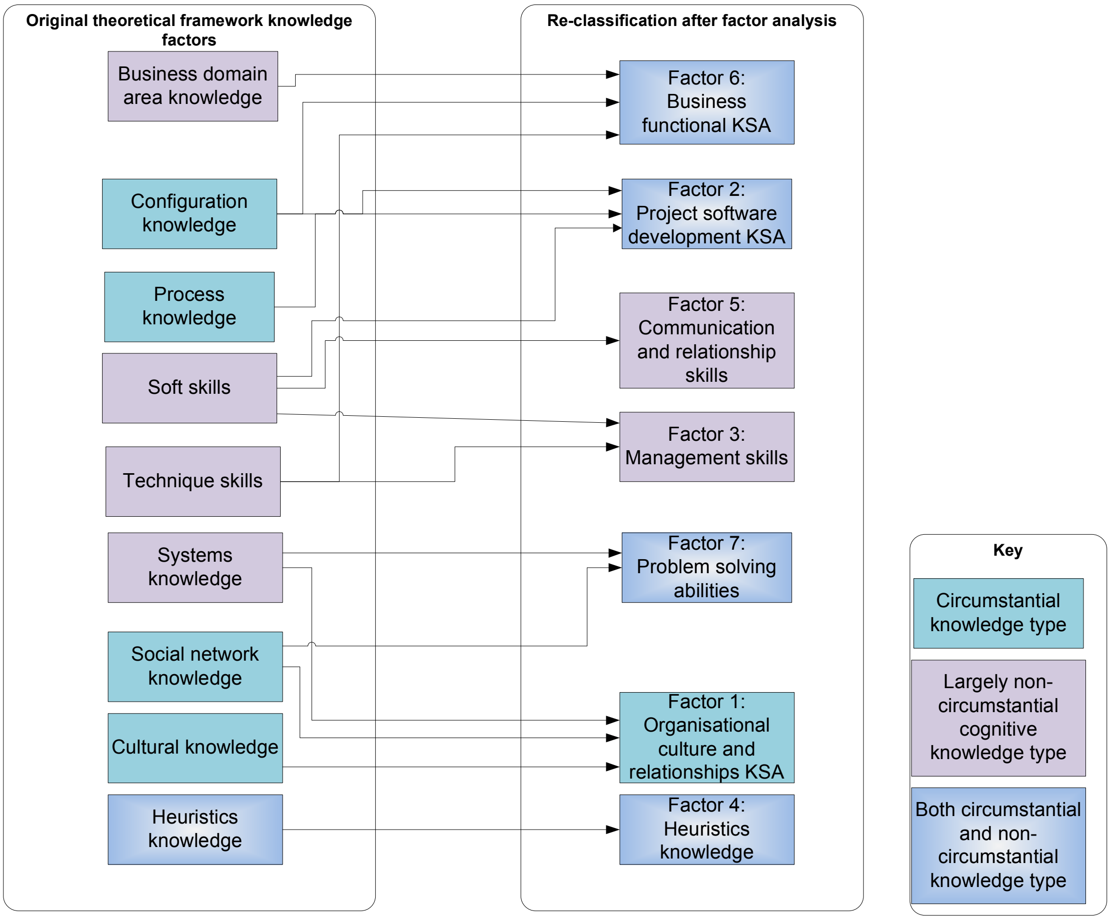

Figure 7.2. Transformation of original theoretical framework

7.3.3 Knowledge requirements for role

EFA revealed seven factors relevant to perform SE roles. The data was further analysed to explore for relationships between role and knowledge factors. Table 7.6 shows the means and standard deviations for the need for different knowledge re-classified factors to perform various roles. The results of ANOVA between role and various identified knowledge factors show that role has statistically significant relationships with Factor 2: Project software development KSA ($p < 0.001$), Factor 5: Communication and relationship skills ($p < 0.05$) (see Table 7.4). Pair-wise comparisons between the different types of roles are reported in Table 7.5.

A statistically significant difference ($p < 0.001$) exists in the need for Factor 2: Project software development KSA between a business analyst and a programmer analyst in performing their job. Programmer-analysts, as per their nature of responsibilities carry out tasks that require higher Factor 2: Project software development KSA. However, Business analysts, typically, do not consider themselves technical (Evans, 2004).

It is found that Factor 5: Communication and relationship skills required for project managers and programmers are also significantly different ($p < 0.024$). This is consistent with the findings in literature that generalist skills such as communication skills were perceived to be significantly more important for managerial employees than for technical employees (Becker et al., 1997). Ives and Olson (1981) found systems managers spend 76% of time in oral communication and therefore need greater communication and relationship skills. The results also demonstrate support to Dollery's (2005) suggestions that a programmer focuses on the programming task at hand for long periods of time to the exclusion of other people related tasks. Hence, it is likely that that there is a statistical difference in this type of skill between a programmer and a project manager.

Source	Dependent Variable	Type III Sum of Squares	df	Mean Square	F	Sig.
A.6Role	x1.Org Culture	11.64	22.00	0.53	1.18	0.28
	x2.Project Software Dev. KSA	17.98	22.00	0.82	2.57	0.00
	x3.ManagementSkills	9.12	22.00	0.41	1.42	0.11
	x4.Heuristics knowledge	11.36	21.00	0.54	1.12	0.33
	x5.Communication and relationship skills	10.61	22.00	0.48	1.68	0.04
	x6.Business Functional KSA	9.52	22.00	0.43	1.21	0.25
	x7.Problem Solving with help	3.290	6	.548	1.766	.110

Table 7.4. ANOVA between role and knowledge factors

Dependent Variable	(I) A6.Role	(J) A6.Role	Mean Difference (I-J)	Std. Error	Sig.	95% Confidence Interval for Difference(a)	
						Lower Bound	Upper Bound
x2.Project Software Dev. KSA	1-Business Analyst	3-Programmer/ Analyst	-.637	.128	.000	-.1.035	.136
X5.Communication and relationship skills	2-Programmer	7-Project Manager	-.723	.180	.023	-1.409	-.036

Table 7.5. Pair-wise comparisons between Roles and Knowledge factors

A6.Role		x1.Org Culture	x2.Software Development KSA	x3.Management Skills	x4.Heuristics	x5.Communication and relations	x6.Business Functional KSA	x7.Problem Solving with help
1-BusinessAnalyst	Mean	2.63	2.60	3.06	2.94	3.47	3.06	3.41
	Std. Devi	0.82	0.72	0.51	0.73	0.48	0.66	0.48
2-Programmer	Mean	2.35	3.05	2.84	2.51	2.99	2.78	2.97
	Std. Devi	0.80	0.56	0.59	0.70	0.60	0.64	0.63
3-Programmer/Analyst	Mean	2.37	3.23	2.76	2.68	3.19	3.01	3.23
	Std. Devi	0.57	0.46	0.54	0.69	0.55	0.52	0.54
4-Architect	Mean	2.30	3.38	2.88	2.67	3.33	3.00	3.17
	Std. Devi	0.32	0.63	0.14	0.58	0.42	0.17	0.63
5-Designer	Mean	2.78	3.10	3.04	2.85	3.52	3.07	3.55
	Std. Devi	0.58	0.76	0.55	0.91	0.54	0.80	0.48
6-Tester	Mean	2.26	2.95	3.15	2.90	3.56	3.17	3.25
	Std. Devi	0.81	0.70	0.53	0.80	0.33	0.66	0.71
7-ProjectManager	Mean	2.83	2.88	3.33	3.06	3.72	3.04	3.27
	Std. Devi	0.60	0.61	0.59	0.59	0.47	0.54	0.54
A1-Other roles	Mean	2.14	2.72	2.77	2.75	3.13	2.55	3.23
	Std. Devi	0.61	0.88	0.50	0.64	0.57	0.76	0.55
A2-Combination roles	Mean	2.26	3.08	2.81	2.41	3.15	2.75	3.08
	Std. Devi	0.75	0.57	0.45	0.60	0.36	0.75	0.56
Total	Mean	2.43	3.04	2.88	2.72	3.26	2.95	3.22
	Std. Devi	0.68	0.62	0.55	0.70	0.56	0.61	0.56

Table 7.6. Knowledge requirements for various roles based on re-classification

7.3.4 Other findings

7.3.4.1 Public/Private sector differences

Respondents from government sector (57.8%) represented a majority than non-government sectors (see section 5.2.2). Tests were conducted to understand the relation between industry sector and the need for knowledge factors. The results are summarised in Table 7.7 and details provided in Table 7.8

	Industry segment	Knowledge factors	Significance (*p*)
1.	Public/Private sector	Factor 2: Software development KSA	0.028
2.	Public/Private sector	Factor 7: Problem solving abilities	0.021

Table 7.7. Relation: Public/Private sector and Need for knowledge factors

Tests of Between-Subjects Effects

Dependent Variable: x2.Software Development KSA

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Corrected Model	1.844[a]	1	1.844	4.927	.028
Intercept	1573.993	1	1573.993	4205.159	.000
A. 11IndustrySeg PublicVsPrivate	1.844	1	1.844	4.927	.028
Error	64.754	173	.374		
Total	1671.936	175			
Corrected Total	66.598	174			

a. R Squared = .028 (Adjusted R Squared = .022)

Tests of Between-Subjects Effects

Dependent Variable: x7.Problem solving abilities

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Corrected Model	1.685[a]	1	1.685	5.467	.021
Intercept	1726.949	1	1726.949	5601.849	.000
A. 11IndustrySeg PublicVsPrivate	1.685	1	1.685	5.467	.021
Error	53.024	172	.308		
Total	1853.792	174			
Corrected Total	54.710	173			

a. R Squared = .031 (Adjusted R Squared = .025)

Table 7.8. Relation: Public/Private sector and need for knowledge factors

Green (1989) found that technical skill requirements were perceived to be of higher need in public sector than private sector. He suggested that such a difference may be due to basic demographic differences (such as age, qualifications) and cultural differences between sectors. This research finds that there is a statistically significant difference in Factor 2: Software development KSA between public sector and private sector. However, it is found that the need for Factor 2 is slightly higher in private sector than in public sector (see Table 7.9). Green (1989) only considered technical skills in his research that are predominantly non-contextual in nature. However, the Factor 2: Software development KSA is inclusive of contextual and non-contextual knowledge types. The difference in findings with Green (1989) could be attributed to this reason. It could also be that times have changed in public sector since Green (1989) carried out his research or that the samples undertaken have different contextual circumstances.

There is also a statistically significant difference in the need for Factor 7: Problem solving abilities with a higher need in public sector than in private sector (see Table 7.9). Problem solving skills require negotiation, ability to resolve conflict and democratic-decision making (Kantrowitz, 2005) and are contextual (Sage and Rouse, 1999). This finding supports the nature of problem solving skills. Both these findings suggest to us that when individuals move between industry sectors they need to be appropriately trained on knowledge types such as Software development KSA and problem solving focussing on its contextual aspects such as knowledge of application system and process knowledge and social network knowledge within the team.

	Mean need for Factor 2: Software Development KSA	Mean need for Factor 7: Problem Solving abilities
Public sector	2.94	3.30
Private sector	3.15	3.10

\# Scale for the importance of a knowledge item to perform the job (knowledge requirements level): 0-not applicable, 1-little help, 2-moderate help, 3-great.4-necessary

Table 7.9. Private/Public sectors: Differences in Knowledge factor needs

7.3.4.2 Relation between demographic profile items and identified knowledge factors

Table 7.10 summarises the significant ($p < 0.05$) relationships found between demographic profiles and the need for knowledge factors to perform the job. Detailed results are available in Appendix 11.7.

	Demographic profile item	Need for Knowledge factors to perform the job	Significance
1.	Defined standards and guidelines	Factor 3:Management skills	0.04
2.	Defined standards and guidelines	Factor 4: Heuristics	0.006
3.	Defined standards and guidelines	Factor 6: Business functional KSA	0.037
4.	Field of study	Factor 2: Software development KSA	0.000
5.	Field of study	Factor 5: Communication and relationship skills	0.020
6.	Gender	Factor 3: Management skills	0.029
7.	Gender	Factor 2: Software development KSA	0.010
8.	Knowledge locator tools	Factor 1: Organisational culture and relationships	0.045

Table 7.10. Relation: Demographic profiles and Need for knowledge factors

Univariate analysis reveals that there is a significant relationship ($p < 0.05$) between organisation having defined standards and guidelines and Factor 3: Management skills. The means table (see section 11.7.1.1 in Appendix) shows that those organisations where there are defined standards and guidelines, there are higher needs for management skills. This can be expected since an organisation having better defined standards and guidelines is likely to have documented procedures and criteria for various phases and tasks in SE work. This means that any coordination needs to follow a methodology set within the organisation. Higher level of knowledge of these documented standards and guidelines and coordination skills is probably required and/or being recognised by managers to perform various jobs.

There is a statistically significant relationship ($p < 0.007$) between defined standards in the organisation and Factor 4: heuristics knowledge (see section 11.7.1.2 for more details). As can be expected, in organisations where there are defined standards and guidelines, the need for Factor 4: heuristics knowledge is also higher. Articulable heuristic knowledge relates to organisational policies and standards and guidelines (Busch et al., 2001). Where such heuristic are well defined, it is possible that individuals make use of them to perform their work. Hence, mean for the need for heuristics knowledge in organisations with defined standards and guidelines (2.789) is higher than organisations with no defined standards and guidelines (2.452).

There is a statistically significant relationship ($p < 0.038$) between defined standards in the organisation and Factor 6: Business functional KSA (see section 11.7.1.3 for more details). In organisations where there are defined standards and guidelines, the need for Factor 6: Business functional KSA is also higher. It is possible that such defined standards and guidelines relate to the business of the organisation and individuals make use of them to perform their work. The mean for the need for Factor 6: Business functional KSA in organisations with defined standards and guidelines (2.989) is higher than organisations with no defined standards and guidelines (2.792). This finding appears to suggest that defined standards and guidelines in organisations are closely aligned with the business functionality.

There is a relationship ($p < 0.021$) between *educational field of study* and the need for Factor 5: Communication and relationship skills (see section 11.7.1.4 for more details). Similarly, there is a relationship ($p < 0.001$) between *educational field of study* and the need for Factor 2: Software development KSA (see 11.7.1.5 for more details). Number of previous research studies on finding knowledge and skill needs have been carried out with a view to improve educational curriculum (Benbasat et al., 1980; Watson et al., 1990a; Lee et al., 1995; Bailey and Stefaniak, 2001). Lemmen (1999) found that education has effects on systems practice. This finding reinforces this finding and suggests that by well-defined educational curriculum, that are developed based on research studies, it is possible to establish measurable differences in the perceived need for different types of knowledge and skills resulting in better outcomes for SE organisations.

The findings regarding relation between gender and Factor 3: Management skills ($p < 0.030$), gender and Factor 2: Software development KSA ($p < 0.031$) appears to be reflective of gender preferences. Women perceive a higher need for Factor 3: Management skills than men to perform their jobs (see 11.7.1.6). On the other hand, men perceive a higher need for Factor 2: Software development KSA to perform their job (see 11.7.1.7). Igbaria and Chidambaram (1997) found significant gender effects on work and life experience, organisational level, salary, boundary spanning activities and intention to stay. There are differences between men and women in terms of listening and communication skills (Brownwell, 2002). Gallivan (2004) found that gender was related to IT professionals' technical skills with women being perceived as having lower technical skills. Gallivan (2004) also found partial support to the proposition that gender was related to IT professionals' non-technical skills with women perceived as having stronger non-technical skills than men. The found relationships re-affirm existing literature that gender preferences indeed are reflective on knowledge and skill requirements. Such preferences need to be taken into account in KM strategies.

There is a significant ($p < 0.046$) relationship between use of knowledge locator tools and the need for Factor 1: Organisational culture and relationships KSA (see 11.7.1.8 for details). Tacit knowledge and organisational culture together with organisational social networks are intertwined (Busch et al., 2001). Knowledge locator tools enable cultural and social network knowledge sharing (Cross and Parker, 2004). Hence, when knowledge locator tools are implemented in the organisation, it is possible that cultural and social network knowledge are needed more in performing various roles. Thus, this finding illustrates the importance of implementing knowledge locator tools in organisations as part of wider KM initiatives.

7.3.4.3 Knowledge contribution for various Software Development Life Cycle (SDLC) phases

Statistical analysis was used to examine the relation between the lengths of time an individual worked on various SDLC phases and need for identified knowledge factors. Significant results ($p < 0.05$) are shown in Table 7.11. Detailed SPSS outputs are presented in Appendix 11.8. The findings are in line with the major tasks that are carried out during the relevant phases. During requirements analysis, it is found in literature that business domain knowledge and application system knowledge along with social network knowledge are important (Walz et al., 1993). But, since no software development is carried out we can expect that Factor 2: Software development KSA is not significant during requirement analysis phase. During design phases, it has been identified in the literature that social network knowledge, communication skills, business domain knowledge and application system knowledge are important (Curtis et al., 1988; Herbsleb and Kuwana, 1993; Walz et al., 1993). The results (see Table 7.11) are in alignment with existing literature.

During 1) architectural design, 2) high level design, 3) low level design, 4) integration testing, 5) production support and 6) configuration management phases, Factor 2: Software development KSA gains importance. During production support phase, both Factor 6: Business functional KSA and Factor 2: Software development KSA are significant. Thus the identified knowledge and skill factors are in alignment with the nature of work that gets carried out in the relevant phases. These results are of importance since students and practitioners need to understand knowledge and skill requirements associated with various SDLC phases (Sulaiman et al., 208). KM strategies need to consider the current and upcoming SDLC phases of the project. Targeted initiatives based on these findings should be undertaken to improve knowledge needs.

Phase	Contributing knowledge factors	Sig. (p)
Requirements analysis	Factor 1: Organisational culture and relationships	0.002
	Factor 3 : Management skills	0.001
	Factor 4 : Heuristics knowledge	0.022
	Factor 5: Communication and relationship KSA	0.000
	Factor 6 : Business functional KSA	0.000
	Factor 7 : Problem solving abilities	0.006
Architectural design	Factor 1: Organisational culture and relationships	0.027
	Factor 2 : Project software development KSA	0.003
	Factor 5: Communication and relationship KSA	0.001
High level design	Factor 1: Organisational culture and relationships	0.049
	Factor 5: Communication and relationship KSA	0.002
Low level design	Factor 2 : Project software development KSA	0.000
	Factor 6 : Business functional KSA	0.011
Coding and unit testing	Factor 2 : Project software development KSA	0.000
Function testing	Factor 6 : Business functional KSA	0.011
Integration testing	Factor 2 : Project software development KSA	0.005
	Factor 6 : Business functional KSA	0.041
Configuration management	Factor 2 : Project software development KSA	0.000
Quality assurance	Factor 1: Organisational culture and relationships	0.047
	Factor 6 : Business functional KSA	0.024
Production support	Factor 2 : Project software development KSA	0.038
	Factor 6 : Business functional KSA	0.036
Preparation of manual	Factor 1: Organisational culture and relationships	0.018
	Factor 3 : Management skills	0.011
Project Management	Factor 3 : Management skills	0.021

Table 7.11. Relation: SDLC and Need for knowledge factors

7.3.4.4 Experience and knowledge requirements

Statistical analysis was carried out between 1) Total Professional Experience and reclassified factors and 2) Experience within the SE team and re-classified knowledge factors to see if there were any significant ($p < 0.05$) relationships between them. The following relationships were found (see Table 7.12 for summary; see detailed output in section 11.9).

- As the Total Professional Experience increases, the Factor 7: Problem solving abilities are needed more to perform their job ($p < 0.017$).
- As the Experience within SE team increases, Factor 5: Communication and relationship skills are needed more to perform the job ($p < 0.049$)

This research captured the knowledge needed to perform the job from respondents. However, if participants possess higher level of knowledge and or abilities of a specific type, then they can be expected to use them in performing their job. It is known in Cognitive psychology literature that past experience usually increases problem solving abilities (Eysenck and Keane, 2005). Hence, it appears, individuals make use of the increased Factor7: Problem solving abilities that is gained as a result of Total Professional Experience.

As individuals work more in a team, they need higher Factor 5: Communication and relationship skills. It has been demonstrated that spontaneous communication is associated with shared identity and shared context (Hinds and Mortensen, 2005). Usually, shared context and shared identity increase as the Experience within SE team increases and hence are closely linked. Hence, it appears that experienced individuals may have gained additional Communication and relationship skills and hence report a higher level of need for such knowledge in performing the job.

Nord and Nord (1997) reported that there were no statistically significant relation between Total experience and perceived knowledge and skill requirements. However, this research results show that there is a statistically significant ($p < 0.016$) relation between Total experience and need for Factor 7: Problem solving abilities. Nord and Nord (1997) study collected 98 responses from experienced systems analysts and less experienced ones. They did not include problem solving skills in their factors. Factor 7: Problem solving abilities includes social network knowledge within the team. This social circumstantial knowledge is not considered as part of the Nord and Nord (1997) research. This problem in the literature possibly explains the reason why Nord and Nord (1997) failed to arrive at a relation between experience and problem solving KSA.

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.
1	(Constant)	3.076	.072		42.788	.000
	Total Professional Experience	.014	.006	.186	2.434	.016

a. Dependent Variable: x7.Problem solving abilities

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.
1	(Constant)	3.174	.058		54.713	.000
	A7.Experience in Software Project team	.031	.016	.148	1.987	.048

a. Dependent Variable: x5.Communication and relationship skills

Table 7.12. Relationship : Experience and skills

7.4 Chapter Summary

This chapter, first, reconciled both case study and general-survey findings and found they were in alignment. Both case study and general-survey research find that all knowledge factors identified in the theoretical framework are required to perform various roles. Both methods find that the increase of the knowledge factors identified in the theoretical framework contribute to improved performance. However, there are differences between case study and general-survey in the rankings of the level of contribution to improved performance. Such a result can be expected since the case study is influenced by the organisational and contextual circumstances. The case study and general-survey have complemented each other in making other findings. For example, the case study findings show that users of the system possess configuration knowledge. Similarly, the general-survey found that the role has a statistically significant relation with perceived contribution of configuration knowledge to increase in performance. EFA was carried out on the data. The EFA revealed seven knowledge factors relevant to perform various SE roles, as follows:

Factor 1: Organisational culture and relationships KSA (Including organisational culture and organisational relationships)

Factor 2: Project software development KSA (Including technical skills, knowledge of application software item in your organisation and processes used in the software project team)

Factor 3: Management skills (Including negotiation, management and organisational skills)
Factor 4: Heuristic knowledge (including standards, guidelines, shortcuts and rules of thumb)
Factor 5: Communication and relationship skills
Factor 6: Business functional KSA (Including business domain area knowledge, user interfaces, business analysis and testing)
Factor 7: Problem solving abilities (Working with others in the team, the ability to investigate, analyse, propose and solve problems and issues)

The research found relations between demographic profile items and need for knowledge factors (examples: relation between industry sector (public/private); gender and knowledge requirements). The analysis found relationships between SDLC phases and the need for knowledge factors. It was found that Total Professional Experience increases the need for Factor 7: Problem solving abilities. Also, it was found that as the Experience within the SE team, increases the need for Factor 5: Communication and Relationship skills to perform the job. These relationships are useful to practice to carry out targeted KM initiatives and are useful to theory.

8 Summary and Conclusions

8.1 Summary

This research project aimed to contribute to theory of individual knowledge and performance for Software Engineering (SE) roles. This contribution to knowledge will be useful to SE practitioners and researchers. Specifically, the following two objectives were set forth for this research:

- Identify the knowledge requirements to perform various SE roles
- Identify the knowledge factors that contribute to performance increase in SE

The literature review demonstrates that the need for 1) technical skills, 2) soft skills, 3) systems knowledge and 4) business domain area knowledge to perform various SE roles has been well recognised. On the other hand, the need for knowledge types 1) social network knowledge, 2) cultural knowledge, 3) configuration knowledge, 4) process knowledge and 5) heuristics knowledge have not been adequately investigated. It was also found from literature that environmental factors such as age, sex, organisational type (government vs private) and organisational maturity can have an effect on the knowledge and skill requirements perceived as necessary to perform the job. However, the existing literature suffers from shortfalls such as 1) No theory based approach to identifying knowledge and skill needs and 2) Inadequate coverage of contextual knowledge.

This research sought to fill this void in the literature by identifying, adapting and applying theories on Knowledge and Action. The thesis investigated cognitive and social origin theories of knowledge and action. This thesis provided literature evidence to demonstrate that SE is indeed a cognitive activity that is carried out in a social context. The research adapted the reconciled cognitive and social origin theories of knowledge and action (Billett, 1995; Billett, 1996; Sveiby, 1997; Shariq, 1999; Packer and Goicoechea, 2000; Augier et al., 2001; Mason, 2007) and the model proposed by Billett (1995). According to this model, both prior cognitive knowledge and knowledge of social circumstances lead to problem solving in a goal directed activity. Participation in goal directed activity results in changed knowledge. This model was applied to SE. A theoretical framework for this research was then formulated, based on a review of the related concepts of competence and operational knowledge, by identifying relevant knowledge types. Based on this framework, the various operational knowledge types identified are as follows:

- Business domain area knowledge
- Configuration knowledge – The knowledge of application software systems.
- Heuristics knowledge
- Process and Procedural knowledge
- Social network knowledge
- Cultural knowledge
- Systems knowledge
- Technique skills
- Soft skills

The research used a positivistic, cross-sectional and exploratory type study, after considering various issues such as suitability of research methods, researcher's philosophical background, capability, and the context of the research relevant to achieving the research objectives. Like numerous other reported research studies on this topic, this study used perception measures, to assess knowledge and its contribution to performance, using a multi-method design as follows:

- A case study that presents a typical view of a real life environment in a software project that answers the research questions
- A perception survey (referred to as general-survey), using non-probabilistic sampling, that provides a snap-shot of experiences of software professionals and answers the research questions.

Quantitative data collected was analysed using descriptive and inferential statistical methods. The rich qualitative data collected was also summarised and provide substantial additional evidence to the quantitative data collected. A reconciliation of general-survey and case study research findings was made. Factor analysis of the general-survey data was carried out to understand the latent structure within the data.

8.2 Conclusions

The research has achieved the objectives set forth with some limitations. The research has uncovered a number of meaningful outcomes which contribute to theory and provide empirical evidence for SE practitioners and researchers. This section presents the main research outcomes and contribution to knowledge.

8.2.1 Knowledge required to perform SE

This research found evidence to demonstrate that both contextual and non-contextual knowledge are required in performing various roles in SE. Thus, this research applies and demonstrates support for the reconciled cognitive and social origin theories of knowledge and action (Billett, 1995; Billett, 1996; Sveiby, 1997; Shariq, 1999; Packer and Goicoechea, 2000; Augier et al., 2001; Mason, 2007). Using both quantitative and qualitative evidence, from both case study and general-survey research, the research found that the knowledge types 1) configuration knowledge (functional and technical knowledge of application system), 2) social network knowledge, 3) process and procedural knowledge, 4) systems knowledge, 5) business domain area knowledge, 6) soft skills, 7) technique skills, 8) cultural and 9) heuristics knowledge are required to perform various SE roles.

The results of the case study appear to show that the technique skills related to the role are directly related to responsibilities of the role (Business analysts requiring business analysis and testing skills, programmers requiring software skills to carry out programming, project managers requiring project management skills). An analysis of configuration knowledge from the case study reveals that programmer-analysts and business analyst/testers need to know some of the application software items in-depth, whereas project managers need to have an overall understanding of all the software items in the system. The results of the case study indicate that programmers require less social network knowledge compared with other roles.

The research presented a number of qualitative examples of different types of knowledge from the case study software project team as well as from the general-survey. Some examples are presented below:

- Cultural knowledge: Corporate practice statements such as IT usage guidelines and ethical guidelines
- Configuration knowledge: Knowledge of the functionality of reports (example: letters issued to clients from the system), batch processes and user interfaces of the application system.

- The importance of social network knowledge in performing the job was highlighted in a statement by a participant that he would call the expert within the team if he did not know.

8.2.1.1 Re-classification

Reclassification of knowledge factors was carried out using EFA. The EFA analysis found seven factors that show close association between contextual and non-contextual knowledge types, demonstrating support to the reconciled cognitive and social origin theories of knowledge and action (Billett, 1995; Billett, 1996; Sveiby, 1997; Shariq, 1999; Packer and Goicoechea, 2000; Augier et al., 2001; Mason, 2007). The research found that the following knowledge types are important to perform various roles in SE:

Factor 1: Organisational culture and relationships Knowledge Skills and Abilities (KSA) (Including knowledge of organisational culture and organisational relationships)

Factor 2: Project software development KSA (Including technical skills, knowledge of application software item in your organisation and processes used in the software project team)

Factor 3: Management skills (Including negotiation, Management and organisational skills)

Factor 4: Heuristic knowledge (including standards, guidelines, shortcuts and rules of thumb)

Factor 5: Communication and team work skills

Factor 6: Business functional KSA (Including business domain area knowledge, user interfaces, business analysis and testing)

Factor 7: Problem solving abilities (Working with others in the team, the ability to investigate, analyse, propose and solve problems and issues)

The research found a number of connections and interrelationships in items within the above factors that support the existing theories in the literature. The structure of Factor 1 demonstrates the interrelationships between tacit knowledge sharing, social network knowledge and culture in an enterprise context and supports theories developed in this regard by Busch et al. (2001). The structure of Factor 2 demonstrates that the non-contextual (example: software skills to develop the application) and contextual knowledge (example: software development process, component functionality) items have a number of interrelationships amongst them and provides substantial evidence to the reconciled cognitive and social origin theories of knowledge and action. The structure of Factor 3 and Factor 5 support the existing literature which found the need for management skills, communication and relationships skills as very important to perform SE roles. The grouping of all heuristic items into Factor 4 provides further empirical evidence to theories on cognitive and social origin theories of knowledge and action (see Chapter 3) which show that experts use heuristics to solve complex irregular problems. Factor 6 provides empirical evidence to Billett's (1995) Model that cognitive skills such as business analysis and testing skills together with contextual knowledge, such as business domain area knowledge and user interfaces, are required to perform SE roles. The structure of Factor 7 demonstrates that analytical problem solving in a goal directed environment is not an individual activity. It is a team activity (Curtis et al., 1988; Rosson, 1996; Cubranic et al., 2004; He, 2004; Ye, 2006). Individuals work together to solve problems. The discovery of this factor appears to show that, in the SE context, the problem solving strategy (Mayer, 1992) is through interactions within the team. This factor shows the interrelationship between non-contextual (investigative skills, analytical skills and problem solving skills) and contextual knowledge items (social network knowledge within the team), thus lending further evidence to the reconciled cognitive and social origin theories of knowledge and action.

It was found that a statistically significant difference exists in the need for Software development KSA between a business analyst and a programmer analyst. As can be expected, due to their nature of the job project managers require higher level of Management skills than programmers/programmer-analysts. Further, perhaps due to the higher needs for communication and establishing relationships (1981) project managers require higher level of communication and relationship skills than programmer-analysts.

8.2.1.2 Knowledge required for SDLC phases

The research identified the significant knowledge requirements for various phases of the software development lifecycle (see Table 7.11 for full list). It was found that during the requirements analysis phase, Factor 2: Software development KSA is not significant. This could be expected since no software development is carried out during this phase. However, during 1) architectural design, 2) high level design, 3) low level design, 4) integration testing, 5) production support and 6) configuration management phases, the Factor 2: Software development KSA gains importance. During design phases, it was identified that social network knowledge, communication skills, business domain knowledge and application system knowledge are important. This is consistent with existing literature (Curtis et al., 1988; Herbsleb and Kuwana, 1993; Walz et al., 1993). During production support phase, both

Factor 6: Business functional KSA and Factor 2: Software development KSA were significant.

8.2.1.3 Experience and knowledge requirements

The research looked at the relation between experience and knowledge factors. It was found that Total Professional Experience increases the need for Factor 7: Problem solving abilities in performing the job. Also, it was found that as the Experience within the project team increases so does the need for Factor 5: Communication and relationship skills in performing the job.

8.2.1.4 Environmental factors and knowledge requirements

An analysis of the relation between demographic profiles and knowledge factors identified significant differences between factors such as gender, field of study and knowledge requirements (see Table 7.10 for full list). It is found that there is a statistically significant difference in Factor 2: Software development KSA between public sector and private sector, with a higher need in private sector (see Table 7.9). There is also a statistically significant difference in the need for Factor 7: Problem solving abilities with a higher need in public sector than in private sector (see Table 7.9). These findings appear to show that contextual and environmental factors influence the knowledge and skill requirements to perform SE roles.

8.2.2 Knowledge factors contributing to performance

Both case study and general-survey research found that all nine of the knowledge factors in the theoretical framework[44] contribute to increase in performance. The final rankings from general-survey are shown in Table 8.1. Consistent with the findings by Rasch and Tosi (1992), it is found that dispositional factors such as 1) achievement needs, 2) effort and 3) self-esteem and 4) personal reasons are also important contributors to increase in performance.

44 1) configuration knowledge (functional and technical knowledge of application system), 2) social network knowledge, 3) process and procedural knowledge, 4) systems knowledge, 5) business domain area knowledge, 6) soft skills, 7) technique skills, 8) cultural and 9) heuristics knowledge.

Type of knowledge	Finalised ranking from survey
Technique skills	1
Configuration knowledge	2
Systems knowledge	3
Business domain knowledge	4
Soft skills	5
Social network knowledge	6
Process knowledge	7
Heuristics knowledge	8
Cultural knowledge	9

Table 8.1. Final rankings

In addition to these findings case study findings show the following:

- New comers have a low level of the following contextual knowledge types:
    - Business domain area knowledge
    - Cultural knowledge
    - Configuration knowledge
    - Social network knowledge
- Users of the system possess necessary configuration knowledge and they are a good source for recruitment for development teams.
- Existing members, who have worked in the organisation before but not on the application system/project team, possess some level of cultural knowledge and social network knowledge.
- Existing members, who have both worked on the application system as well as in the organisation, report some levels of cultural, social network and configuration knowledge.

In addition, the general-survey research revealed several relationships between demographic profile items and perceived contribution of increase in knowledge to increase in performance. These are reported below[45]:

- It was found that as the Total Professional Experience increases, the perceived contribution of process knowledge, general systems knowledge and cultural knowledge to improvements in performance declines.

45 Increase in knowledge and improvement in performance since joining the SE team

- As the Experience within SE team increases there is higher perceived contribution of increase in technique skills and configuration knowledge to increase in performance.
- There is a relation between role and perceived contribution of increase in configuration knowledge to increase in performance. The contribution is higher for programmers, programmer-analyst, designers and architects. It is comparatively lower for project managers, testers and business analysts.

Thus, this research has advanced theory by applying and demonstrating theories of cognitive and social origins of knowledge and action in SE. The research found seven types of knowledge factors that are required to perform various roles in SE. The research ranked the knowledge types in terms of their contribution to improvements in performance and identified knowledge types that are required during various SDLC phases. Further the research has found relationships between demographic profiles and knowledge factors required to perform various roles. The research found the relation between experience and the need of knowledge types. These are substantive advancements to knowledge and rectified the shortfalls in the literature including 1) lack of theory based approaches to identifying knowledge and skill needs and 2) Inadequate coverage of contextual knowledge.

8.2.3 Implications for theory

Cognitive and social origin theories offer a substantial theoretical basis for explaining the knowledge and performance of individuals carrying out various roles in SE. This research adapted, applied and found empirical support for reconciled cognitive and social origin theories of knowledge and action (Billett, 1995; Billett, 1996; Sveiby, 1997; Shariq, 1999; Packer and Goicoechea, 2000; Augier et al., 2001; Mason, 2007). The thesis arrived at a new framework for knowledge requirements to perform various SE roles, thus building a theory for analysing and describing (Gregor, 2006). Thus, this thesis has advanced theory by applying the theories of cognitive and social origins of knowledge into SE and providing empirical evidence. Few examples in the literature surveyed on knowledge/skill requirements for SE have been based on any sound fundamental theories of knowledge and action (see section 2.2.5.4). The research provides substantial contribution to knowledge by testing the proposed theories in a real life case study example as well as through a general-survey. Thus, applying Gregor's (2006) assessments, by arriving at a classification theory for analysing and describing, that is founded upon sound theories from psychology literature, the research contributes to knowledge.

Further, the identified statistical relations between knowledge needs and factors such as 1) SDLC phases, 2) Experience, 3) demographic profile factors (such as gender) are new findings. Even though the analysis is exploratory in nature, nevertheless they represent a new discovery and hence are a substantial contribution to theory. Scientific knowledge can always be advanced through further formulation and empirical testing of theories. Through its findings, this research has opened up new avenues into the understanding of SE knowledge and performance.

8.2.4 Implications for practice

Various knowledge types required to perform SE roles have been identified in this research. An important finding is that both contextual knowledge as well as non-contextual cognitive knowledge is required to perform various roles in SE. Further, the research found that all the identified knowledge types are important for performance increase. The research was able to arrive at a ranking of knowledge factors based on the contribution of its increase to increase in performance. The implications for practice are discussed in the sub-sections.

8.2.4.1 Better Knowledge Management

SE is a knowledge intensive activity where the end product and input raw material are both knowledge. Hence, the thesis argues that an underlying cause of failure is knowledge, skills and abilities of individuals involved. It is likely that a better knowledge perspective of the participants of the project and better knowledge management initiatives to address the knowledge and skill needs of the participants could reduce such project failures. Several such KM initiatives have been described in the literature (Corbin et al., 2007; Aurum et al., 2008; Bjornson and Dingsoyr, 2008; Dingsoyr et al., 2009). The research addressed the question of knowledge requirements to perform various roles in SE. The identified knowledge types stand as the basis for developing KM strategies. Further, the research has identified 1) the knowledge needed for various SDLC phases, 2) relation between knowledge experience and knowledge needs 3) importance of contextual knowledge 4) relation between knowledge needs and demographic profile factors and industry sectors.

Based on these knowledge requirements and identified relations there are opportunities for industry to implement appropriate KM initiatives to reduce failures, improve productivity and improve overall management. For instance, it is possible to carry out focussed KM initiatives for each of the SDLC phases based on the results of this research. When individuals move between public and private sectors they need to be appropriately trained in 1) Software development KSA and 2) Problem solving abilities focussing on their contextual aspects such as a) knowledge of application system and b) process knowledge and c) social network knowledge within the team. Similarly, gender differences (Women perceiving a higher need for Factor 3: Management skills and men perceiving a higher need for Factor 2: Software Development KSA) need to be taken into account in KM strategies. It is found that implementation of knowledge locator tools improves tacit knowledge such as cultural and social network knowledge needs. The literature shows that KM initiatives in SE have a general lack of focus on tacit knowledge management (Bjornson and Dingsoyr, 2008). The results of this study demonstrate the need to further carry out effective tacit knowledge management through strategies based on the cartographic perspective[46]. Therefore, it is recommended that organisations make use of knowledge locator tools as part of broader KM initiatives. It is found that in organisations where there are defined standards and guidelines, there are higher need for management skills and heuristic knowledge. Hence, organisations aiming to implement such schemes need to introduce strategies to impart this knowledge through targeted KM initiatives.

The research ranked knowledge types based on their contribution to improvements in performance. The research found support for a number of relationships between demographic variables such as Total Professional Experience and Experience within the project team. Technique skills and Configuration knowledge were ranked as top knowledge types contributing to increase in performance. Hence, KM initiatives should be targeted to improving these types of knowledge so that performance gains can be obtained.

The results of the study are helpful as a foundation for competency analysis. Organisations should carry out a knowledge needs analysis as part of competency management process (Nelson, 1991; Davenport, 1997; Curtis et al., 2003). This analysis can be carried out by a knowledge management expert together with relevant stakeholders. The theoretical framework proposed and tested in this research can act as the foundation for this needs analysis. This analysis will shed light on the knowledge requirements for each role and help identify appropriate knowledge management strategies can be developed and implemented.

---

46 focuses on knowledge maps and creating knowledge directories (Bjornson and Dingsoyr, 2008).

8.2.4.2 Recruitment, attrition and personnel rotation

The findings from this thesis on knowledge and skill requirements will help managers to develop and use appropriate strategies during recruitment, attrition and personnel rotation. For example, this thesis recommends important knowledge types to perform various roles in SE. During recruitment, HR managers will need to focus on these types of knowledge requirements. Similarly, during personnel rotation, managers will need to understand the knowledge requirements of each position and take appropriate strategies to ensure that the required knowledge is held by the incoming person to the position.

8.2.4.3 Project management and knowledge

This research finds that both contextual and non-contextual knowledge are required to perform various SE roles. Project managers and systems managers in general should understand that both circumstantial or contextual and non-contextual skills are important to perform various SE roles. People, who hold knowledge, are not simply resources that can be replaced. It is found from the case study that outsiders from organisations generally have a low level of most of the contextual knowledge types. Even if we can find an outsider with appropriate non-contextual cognitive skills to perform various roles, it takes time for them to gain contextual knowledge required to perform the roles within the SE teams. Senior management and Project managers should be cognisant of this fact and take measures to retain and manage contextual knowledge within their teams and organisations.

8.2.4.4 Better curriculum design

The outcomes of this research have significance for curriculum design. It is found that through educational curriculum developed based on research studies, it is possible to establish measurable differences in the perceived need for different types of knowledge and skills to perform SE roles. This can result in better outcomes for SE organisations. Curriculum designers should invest resources in areas which are perceived as more important for the profession (Tye et al., 1995). Hence, curriculum development needs to incorporate and emphasise the importance of contextual knowledge and teach various strategies for individuals to develop contextual knowledge. For example, curricula should stress the importance of application systems knowledge and business domain knowledge and present strategies on how such knowledge can be acquired quickly while working in a team environment. Industry bodies should emphasise the importance of contextual knowledge in formulating core Body of Knowledge requirements, such as SWEBOK.

8.2.5 Limitations of the research

As with all research, there is a number of weaknesses in this research. These are discussed in this sub-section.

Perception only

This research collects the perceived knowledge requirements and perceived performance requirements for performing various software development roles. The perceptions show us an inside information on what individuals think. However, it may not be the real requirement for performing their jobs. The study only collects information on the perceived knowledge and perceived performance and factors contributing to increase in performance of participants. The research question is about finding the knowledge requirements for performing a various jobs. A critique could argue that perceptions for performing the current job may not be the same as the actual knowledge requirements to perform that role. This criticism is a valid one. The characteristics of knowledge, such as its tacitness, being action oriented (Sveiby, 1997), make it difficult to measure objectively by observation or through other means and find the knowledge requirements to perform the role. The only available option is to obtain the perceptions of individuals on the knowledge required to perform their jobs.

Inarticulable tacit knowledge is not measured

Knowledge is not a material commodity that can be measured easily. Polanyi (1958, cited in Sveiby, 1997) says that knowledge is mainly tacit. Polanyi (1958 cited in Sveiby, 1997) regards knowledge as personal and formed within a social context. Individuals may use the tacit part of their knowledge to perform their jobs. They may not be aware that they know some thing and use them to perform their job. In such cases participants would not explicitly elaborate on the tacit knowledge components as much as the explicit components of knowledge. Such inarticulable tacit knowledge (Busch et al., 2001) cannot be captured using a perception survey.

Statistical limitations

The sampling methodology used in this study is a purposive-judgemental sampling approach. This is a non-probabilistic sampling methodology. The results of the study can be generalised to theory with the fact that the sampling is non-probabilistic and may be biased. Further, an item to sample ratio of 1: 3.56 for factor analysis purposes is low.

The study uncovered number of statistically significant ($p < 0.005$) relationships. However, the R-square value for the reported relationships are generally lower (R-square < 0.1). R-square can be lower in case of situations where there are wide variations reflecting respondents opinions and does not necessarily mean no meaningful relationships exist in data (Colton and Bower, 2008). The Princeton University statistical services (2008) says that R-square is generally of secondary importance unless the main concern is to make accurate predictions using regression equation . The underlying data for this research reflects wide ranging respondent's perceptions about knowledge and performance and variation can be expected. The relationships discovered in this research are exploratory in nature and not intended for accurate predictions. Hence, even with small R-square values, the discovered relationships provide practical importance to theory and practice.

Overlaps in roles and positions

Different positions with same role title, within different organisations (even within the same organisation) involve different requirements for knowledge and may involve substantial variation in job specifications. Any generalised theoretical framework will not apply to a given position straightway. A knowledge needs analysis should be performed with the available theoretical framework and the knowledge requirements should be identified.

Real and ideal knowledge needs

The study collects opinions on knowledge requirements for performing various roles and performance increases from the participants. Based on the data collected, the study arrives at the knowledge requirements for performing the various roles. However, the perceived knowledge requirements from individuals could be different from the real knowledge requirements for the job. Also, there may have been an ideal set of knowledge requirements to perform these roles, which is not known or not practised by the professionals in the industry. The research has a limitation in that aspect since it only collects the perceived knowledge requirements to perform their job from practitioners in software industry. The general-survey also does not differentiate by rating the performance of the practitioners.

Static in time

The study collects opinions at a certain point in time and reports the results. The knowledge requirements and performance increases can change in the future due to various reasons (examples: technological change, changes to role definitions). The study cannot predict the future. Any variations in future must be researched and reported at that time.

8.2.6 Possible future work

There are several opportunities for future work that can emanate from this research. The following list some of the author's high level suggestions.

8.2.6.1 Measurement of knowledge and performance through other means

The research measured knowledge and performance through perceptions of individuals. It may be possible to operationalise and measure knowledge using different measures specific to roles[47] rather than perceptions. Future research should investigate the possibilities of using such measures to further demonstrate the underlying theories.

8.2.6.2 Test additional findings from case study

It was found from case study results, that different categories of team members possess different levels of knowledge. For example, newcomers posses little or no contextual knowledge, while those who worked in the organisation previously possess some contextual knowledge. Users of the application system possess configuration knowledge. These findings can be tested in a large-scale study in the future.

8.2.6.3 Develop, implement and test Knowledge Management strategies

New research can be carried out to develop, implement and analyse the effectiveness of KM strategies. Action research in real life circumstances can be carried out to understand the real improvements to performance due to changes in knowledge of different knowledge types.

8.2.6.4 Modelling of SE team knowledge

Efforts can be made to model SE team knowledge, including factors such as attrition rate, recruitment rate. These models can be tested in a simulated environment to understand the knowledge flows and needs at any point of time. Efforts can also be made to model real life environments. Such models will be of immense use to the industry at large.

8.2.6.5 Codification or Personalisation strategy for SE

This research has identified the knowledge requirements for SE. More detailed studies can be carried out to understand the optimum balance between codification and personalisation strategies (Tiwana, 2000) for KM in SE.

8.2.6.6 Longitudinal studies on SE

Longitudinal studies can be performed to prove causal connection between knowledge and performance. Such studies may be able to prove the relation, using a time series approach, the impact of knowledge on the action of an individual.

[47] or tasks

8.2.7 Final remarks

SE is filled with a number of failure and success stories. While being considered as an engineering activity, SE is essentially a social process that revolves around people, business functionality, technology, processes and organisational culture. In order for someone to effectively carry out SE activities, individuals need to be proficient in technique skills as well as the knowledge of social circumstances including business functionality and the application system. Senior executives, systems managers and SE community including academicians need to understand the importance of contextual knowledge to performance. When an individual with contextual knowledge leaves the organization or project team, contextual knowledge walks out the door and cannot easily be replaced. While technical, business or systems knowledge may be found in individuals outside the organization, the organisation will not be able to replace the full scope of that individual's knowledge, skills and abilities. There are several implications here for organizations. First, it is important to know the full scope of KSAs within a project team, both in terms of what the project needs and what KSAs the project team members have between them. This enables the organisation to identify what skills and knowledge are being lost and to work out the best ways of replacing them. Secondly, if contextual knowledge was an important element of a leaving individual, then it is unlikely that this can be replaced from an individual recruited from outside. On the other hand, such knowledge grows with exposure and experience and can be contributed from other members within the project team. In addition, this is the kind of knowledge an organisation can build in team members, requiring ways of education or training for new team members. People should not be treated merely as resources that can be easily replaced; instead measures should be taken to build and retain contextual knowledge for effective SE.

8.2.7.1 Postscript: Personal Reflection

This research project presented the author with a rare opportunity to carry out in-depth study in SE and KM. The author was only a practitioner of SE before embarking on this project. During the last eight or more years of study, the author of this research has gained substantial knowledge in a number of disciplines, above all the art and science of research. Through this journey, the author has gained perspective, experience and insight to look at the world from a scientific vantage point.

9 References

ABS (2004). Information and Communication Technology, ABS.

ACS. (2008). "ICT Careers Portal." Retrieved 10th Jan 2008, 2008, from **http://www.acs.org.au/ictcareers/index.cfm?action=list&sgrID=200707131235515720**.

Adelson, B. and E. Soloway (1985). "The Role of Domain Experience in Software Design." IEEE Transactions on Software Engineering 11(11).

Alagarsamy, K., S. Justus, et al. (2007). On the Implementation of a Knowledge Management Tool for SPI. International Conference on Computational Intelligence and Multimedia Applications, IEEE.

Alavi, M. and P. Carlson (1992). "A Review of MIS Research and Disciplinary Development." Journal of Management Information Systems 8(4): 45.

Althoff, K.-D., F. Bomarius, et al. (2000). "Knowledge Management for Building Learning Software Organizations." Information Systems Frontiers 2(3/4): 349-367.

Anderson, J. R. (1982). "Acquisition of cognitive skill." Psychological Review 89(4): 369-406.

Anderson, J. R. (1993). "Problem solving and learning." American Psychologist 48(1): 35-44.

Anklam, P. (2003). "KM and the social network." Retrieved 8 May 20034, 2004, from **http://www.kmmagazine.com/xq/asp/articleid.F79B4E31-7854-4B6A-9202-164FB18672D3/qx/display.htm.**

Arent, J. and J. Nrbjerg (2000). Software Process Improvement as Organizational Knowledge Creation: A Multiple Case Analysis. Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 4 - Volume 4, IEEE Computer Society.

Atman, C. J., R. S. Adams, et al. (2007). "Engineering design processes: A comparison of students and expert practitioners." Journal of Engineering Education 96(4): 359-379.

Augier, M., S. Shariq, et al. (2001). "Understanding context: its emergence, transformation and role in tacit knowledge sharing." Journal of Knowledge Management 5(2): 125-137.

Aurum, A., F. Daneshgar, et al. (2008). "Investigating Knowledge Management practices in software development organisations - An Australian experience." Inf. Softw. Technol. 50(6): 511-533.

Australian Computer Society. (2008). "ICT Careers Portal : Business Analyst." Retrieved 10th Jan 2008, 2008, from

http://www.acs.org.au/ictcareers/index.cfm?action=show&conID=businessanalyst.

Awad, E. and H. Ghasiri (2003). Knowledge Management. New Delhi, Pearson Education (Singapore) Indian Branch.

Bailey, J. and G. Stefaniak (2001). Industry perceptions of the knowledge, skills, and abilities needed by computer programmers. Proceedings of the 2001 ACM SIGCPR conference on Computer personnel research. San Diego, California, United States, ACM.

Bandura, A. (1986). Social Foundations of Thought and Action : A Social Cognitive Theory. Englewood Cliffs, NJ., Prentice-Hall, Inc., .

Bansler, J. and K. Bodker (1993). "A reappraisal of structured analysis: design in an organizational context." ACM Trans. Inf. Syst. 11(2): 165-193.

Barone, D. F., C. R. Snyder, et al. (1997). Social cognitive psychology : history and current domains. New York, Plenum Press.

Barribeau, P., B. Butler, et al. (2005). "Survey Research. Writing@CSU." 15 Aug 2007, from **http://writing.colostate.edu/guides/research/survey/contrib.cfm**.

Barthelmess, P. and K. M. Anderson (2002). "A View of Software Development Environments Based on Activity Theory." Computer Supported Cooperative Work 11: 13.

Baskerville, R. and J. Pries-Heje (1999). "Knowledge capability and maturity in software management." SIGMIS Database 30(2): 26--43.

Baskerville, R., J. Pries-Heje, et al. (2007). "The enduring contradictions of new software development approaches: a response to 'Persistent Problems and Practices in ISD'." Information Systems Journal 17(3): 241-245.

Bassellier, G. and I. Benbasat (2004). "Business competence of information technology professionals: Conceptual development and influence on it-business partnerships." MIS Quarterly 28(4): 673-694.

Beazley, H., J. Boenisch, et al. (2002). Continuity Management : Preserving Corporate Knowledge and Productivity when employees leave. Hoboken, N.J., John Wiley & Sons, Inc.

Beazley, H., J. Boenisch, et al. (2003). "Knowledge continuity: The new management function." Journal of Organizational Excellence 22(3): 65-81.

Becker, J., R. Insley, et al. (1997). "Communication skills of technical professionals: a report for schools of business administration." SIGCPR Comput. Pers. 18(2): 3-19.

Benbasat, I., A. Dexter, et al. (1980). "Impact of organizational maturity on information system skill needs." MIS Quarterly 4(1): 21.

Benbasat, I., D. K. Goldstein, et al. (1987). "The Case Research Strategy in Studies of Information Systems." MIS Quarterly 11(3): 368.

Benson, S. and C. Standing (2001). "Effective Knowledge Management : Knowledge Thinking and the Personal-Corporate Knowledge Nexus Problem." Information Systems Frontiers 3(2): 227.

Billett, S. (1995). Ph.D Thesis, Structuring knowledge through authentic activities. Faculty of Education, Griffith University.

Billett, S. (1996). "SITUATED LEARNING: BRIDGING SOCIOCULTURAL AND COGNITIVE THEORISING." Learning & Instruction 6(3): 263-280.

Billett, S. (2001). "Knowing in practice: re-conceptualising vocational expertise." Learning & Instruction 11(6): 431-452.

Bjornson, F. O. and T. Dingsoyr (2008). "Knowledge management in software engineering: A systematic review of studied concepts, findings and research methods used." Information And Software Technology 50(11): 1055-1068.

Blanton, J. E., T. Schambach, et al. (1998). "Factors affecting professional competence of information technology professionals." SIGCPR Comput. Pers. 19(3): 4-19.

Boehm, B. W. (1981). Software Engineering Economics. Englewood Cliffs, NJ, Prentice-Hall, Inc.

Bogoiavlenski, I. A., A. G. Clear, et al. (1997). "Historical perspectives on the computing curriculum." SIGCUE Outlook 25(4): 94-111.

Boisot, M. (1998). Knowledge Assets: Securing competitive advantage in the information economy. Oxford, Oxford University Press, U.K.

Borchers, G. (2003). The software engineering impacts of cultural factors on multi-cultural software development teams. Proceedings of the 25th International Conference on Software Engineering. Portland, Oregon, IEEE Computer Society.

Borgatti, S. and J. Molina (2003). "Ethical and Strategic Issues in Organizational Social Network Analysis." Journal of Applied Behavioral Science 39(3): 337-349.

Brown, M. G. (2000). Winning Score : How to design and implement organizational scorecards, Productivity Press.

Brownwell, J. (2002). Listening Attitudes, Principles and Skills. Sydney, Allyn and Bacon.

Buckingham, A. and P. Saunders (2004). The survey methods workbook. Cambridge, UK, Polity press ltd.

Bukowitz, W. and R. L. Williams (1999). The Knowledge Management field book. London, Pearson Education Limited.

Burns, R. (2000). Introduction to Research Methods. Thousand Oaks, CA, Sage Publications.

Busch, P. A., D. Richards, et al. (2001). Visual mapping of articulable tacit knowledge. Australasian Symposium on Information Visualisation, Sydney, Australian Computer Society.

Butler, T. O. M. and B. Fitzgerald (2001). "The Relationship Between User Participation and the Management of Change Surrounding the Development of Information Systems: A European Perspective. (Industry Trend or Event)." Journal of End User Computing 13(1): 12.

Byrd, T. A., K. L. Cossick, et al. (1992). "A synthesis of research on requirements analysis and knowledge acquisition techniques." MIS Quarterly 16(1): 117.

Campbell, J. (1997). "The impact of video conferenced meetings on the pattern and structure of organisational communication." Singapore Management Review 19(1).

Canfora, G., A. Cimitile, et al. (2002). From Knowledge Management concepts toward software engineering practices. Product Focused Software Process Improvement, Proceedings. 2559**:** 407-422.

Cash, E., P. Yoong, et al. (2004). "The impact of e-commerce on the role of IS professionals." SIGMIS Database 35(3): 50-63.

Cavana, R., Delahaye, BL & Sekaran, U (2001). Applied Business Research: Qualitative and Quantitative Methods. Australia, John Wiley & Sons.

Chan, R. (2003). Ph.D Thesis, Structuring and modelling knowledge in the context of enterprise systems. Centre for Information Technology Innovation, Queensland University of Technology.

Cheney, O. H. (1988). Information systems skills requirements: 1980 & 1988. Proceedings of the ACM SIGCPR conference on Management of information systems personnel. College park, Maryland, United States, ACM Press.

Cheney, P. H. and N. R. Lyons (1980). "Information systems skill requirements : A survey." MIS Quarterly 4(1): 35.

Child, D. (1990). The essentials of factor analysis. London, Cassel Educational Limited.

Chung, W. Y. and P. J. Guinan (1994). Effects of participative management on the performance of software development teams. Proceedings of the 1994 computer personnel research conference on Reinventing IS : managing information technology in changing organizations: managing information technology in changing organizations. Alexandria, Virginia, United States, ACM.

Collis, J. H., R (2003). Business Research: A Practical Guide. Great Britain, Palgrave Macmillan.

Colton, J. and K. Bower. (2008). "Some Misconceptions about R-square." Retrieved 31st Dec 2008, 2008, from **http://www.minitab.com/resources/Articles/MisconceptionsR2.pdf**.

Cooke-Davies, T. (2002). "The Real Success Factors on Projects." International Journal of Project Management 20(3): 185-190.

Corbin, R. D., C. B. Dunbar, et al. (2007). "A three-tier knowledge management scheme for software engineering support and innovation." Journal Of Systems And Software 80(9): 1494-1505.

Costello, A. B. and J. W. Osborne (2005). "Best Practices in Exploratory Factor Analysis: Four Recommendations for Getting the Most From Your Analysis." Practical assessment research and evaluation : A peer reviewed electronic journal 10(7).

Cross, N. (2004). Expertise in design: an overview, Elsevier Sci Ltd.

Cross, R. and A. Parker (2004). The Hidden Power of Social Networks. Boston, Massachusetts, Harvard Business School Press.

Cubranic, D., G. Murphy, et al. (2004). Learning from project history: a case study for software development. Proceedings of the 2004 ACM conference on Computer supported cooperative work. Chicago, Illinois, USA, ACM.

Curtis, B., B. Hefley, et al. (2003). "People Capability Maturity Model (P-CMMM) version 2.0." Retrieved 20 Nov 2007, 2007, from **http://www.sei.cmu.edu/publications/documents/01.reports/01mm001.html**.

Curtis, B., H. Krasner, et al. (1988). "A field study of the software design process for large systems." Commun. ACM 31(11): 1268--1287.

Curtis, B., D. Walz, et al. (1990). Studying the process of software design teams. Proceedings of the 5th international software process workshop on Experience with software process models. Kennebunkport, Maine, United States, IEEE Computer Society Press.

Darke, P. and G. Shanks, Eds. (2000). Case study research. Research methods for students and professionals : Information management and systems. Wagga Wagga, Centre for information studies, Charles Sturt University, New South Wales.

Davenport, T. (1997). "Knowledge Management Case Study : Knowledge Management at Microsoft, 1997." 20 Sep 2007, from **http://www.itmweb.com/essay536.htm**.

Davenport, T. and L. Prusak (1998). Working Knowledge : How Organizations manage what they know ? Boston, MA., Harvard Business School Press.

de Souza, C. R., S. Quirk, et al. (2007). Supporting collaborative software development through the visualization of socio-technical dependencies. Proceedings of the 2007 international ACM conference on Supporting group work. Sanibel Island, Florida, USA, ACM.

Defence Signals Directorate. (2008). "Australian Government ICT security manual (ISM)." Retrieved 17 April 2009, 2009, from **http://www.dsd.gov.au/library/infosec/ism.html**.

DeLone, W. H. and E. McLean (1992). "Information Systems Success: The Quest for the Dependent Variable." Information Systems Research 3(1): 60-95.

DeMarco, T. (1982). Controlling Software Projects : Management, Measurement and Estimation. New York, NY, Yourdon Press.

Desouza, K. C. (2003). "Barriers to effective use of knowledge management systems in software engineering." Communications of the ACM 46(1): 99-101.

Devanbu, P. and R. J. Brachman (1991). LaSSIE: A knowledge-based software information system. Communications of the ACM, Association for Computing Machinery. 34**:** 34.

Dictionary.com. (2007, 15 Nov 2007). "Performance. (n.d)." from **http://dictionary.reference.com/browse/performance**.

Dingsoyr, T. (2002). Ph.D Thesis, Knowledge management in medium-sized software consulting companies. Computer Science, Norges Teknisk-Naturvitenskapelige Universitet, Norway**:** 256.

Dingsoyr, T., F. O. Bjarnson, et al. (2009). "voice of evidence What Do We Know about Knowledge Management? Practical Implications for Software Engineering." IEEE Software 26(3): 100-103.

Dingsoyr, T. and R. Conradi (2002). "A survey of case studies of the use of knowledge management in software engineering." International Journal Of Software Engineering And Knowledge Engineering 12(4): 391-414.

Dixon, N. (2000). Common Knowledge: How companies thrive by sharing what they know. Boston, MA, Harvard business school press.

Dokko, G., S. L. Wilk, et al. (2009). "Unpacking Prior Experience: How Career History Affects Job Performance." Organization Science 20(1): 51-68.

Dollery, B. (2005). "Understanding the Psychology of Programming." Retrieved 20 Dec 2007, 2007, from **http://www.devx.com/DevX/Article/11659/1954?pf=true**.

Downey, J. (2005). A framework to elicit the skills needed for software development. SIGMIS CPR '05: Proceedings of the 2005 ACM SIGMIS CPR conference on Computer personnel research, New York, NY, USA, ACM Press.

Downey, J. (2006). Systems architect and systems analyst: are these comparable roles? Proceedings of the 2006 ACM SIGMIS CPR conference on computer personnel research: Forty four years of computer personnel research: achievements, challenges \& the future. Claremont, California, USA, ACM.

Downey, J. and N. Power (2007). An artefact-centric framework for software development skills. SIGMIS-CPR '07: Proceedings of the 2007 ACM SIGMIS CPR conference on 2007 computer personnel doctoral consortium and research conference, New York, NY, USA, ACM Press.

Dyba, T. (2005). "An empirical investigation of the key factors for success in Software Process Improvement." IEEE Transactions on Software Engineering 31(5): 410-424.

Ehrlich, D. (2004). "HRD 408: Instructional Design II." Retrieved 18th Oct 2004, 2004, from **http://www.neiu.edu/~dbehrlic/hrd408/glossary.htm#top**.

Ericsson, K. A. and A. C. Lehmann (1996). "Expert and exceptional performance: Evidence of maximal adaptation to task constraints." Annual Review of Psychology 47: 273-305.

Evans, N. (2004). "The Need for an Analysis Body of Knowledge (ABOK) - Will the Real Analyst Please Stand Up ?" Retrieved 1 Jan 2009, 2009, from **http://proceedings.informingscience.org/InSITE2004/053evans.pdf.**

Ewusi-Mensah, K. (2003). Software Development Failures. Boston, MA, USA, MIT Press.

Eysenck, M. and M. Keane (2005). Cognitive Psychology. New York, Psychology Press.

Fischer, G. (1996). Domain-Oriented Design Environments. 18th International Conference on Software Engineering (ICSE'96), IEEE.

Fischer, G. and J. Ostwald (2001). "Knowledge Management: Problems, Promises, Realities and Challenges." IEEE Intelligent systems: 60-72.

Fitzgerald, B. (1997). "The use of systems development methodologies in practice: a field study " Information Systems Journal 7(3): 201-212(212).

Forgas, J. P. (1981). Social cognition : perspectives on everyday understanding. London ; New York, Published in cooperation with European Association of Experimental Social Psychology by Academic Press.

Forward, A. and T. Lethbridge (2002). The relevance of software documentation, tools and technologies: a survey. Proceedings of the 2002 ACM symposium on Document engineering. McLean, Virginia, USA, ACM.

Frampton, K., J. Thom, et al. (2006). Information technology architects: approaching the longer view. Proceedings of the 2006 ACM SIGMIS CPR conference on computer personnel research: Forty four years of computer personnel research: achievements, challenges \& the future. Claremont, California, USA, ACM Press.

Freeman, L. A. (2001). "Information Systems Knowledge: Foundations, Definitions, and Applications." Information Systems Frontiers 3(2): 249.

Gallivan, M. (2004). "Examining IT professionals' adaptation to technological change: the influence of gender and personal attributes." SIGMIS Database 35(3): 28-49.

Gallivan, M. J., D. Truex III, et al. (2004). "Changing patterns in IT skill sets 1988-2003: a content analysis of classified advertising." SIGMIS Database 35(3): 64-87.

Garson, D. (1998). "Factor analysis." Retrieved 15th Aug 2007, 2007, from **http://www2.chass.ncsu.edu/garson/pa765/factor.htm**.

Garson, D. (2007). "Factor Analysis." Retrieved 20 Dec 2007, 2007, from **http://www2.chass.ncsu.edu/garson/pA765/factor.htm**.

Gasson, S. (1998). Framing design: a social process view of information system development. Proceedings of the international conference on Information systems. Helsinki, Finland, Association for Information Systems.

Gasson, S. (1999). "A social action model of situated information systems design." SIGMIS Database 30(2): 82-97.

Gigerenzer, G. and P. M. Todd (1999). Simple heuristics that make us smart. Oxford, Oxford University Press.

Glass, R. (1997). "The ups and downs of programmer stress." Commun. ACM 40(4): 17-19.

Glass, R. L. (1996). "The relationship between theory and practice in software engineering." Commun. ACM 39(11): 11-13.

GMU. (2002). "Notes on Nicomachean ethics book VI." Retrieved 15 April 2009, 2009, from **http://www.gmu.edu/courses/phil/ancient/nevi.htm**.

Green, G. I. (1989). "Perceived Importance of Systems Analysts' Job Skills, Roles, and Non-Salary Incentives." MIS Quarterly 13(2): 114-133.

Gregor, S. (2006). "THE Nature of theory in Information Systems." MIS Quarterly 30(3): 611-642.

Guinan, P. and R. Bostrom (1986). "Development of Computer-Based Information Systems : A communication framework." Data Base 17(3): 3-16.

Guinan, P. J., J. G. Cooprider, et al. (1998). "Enabling Software Development Team Performance During Requirements Definition: A Behavioral Versus Technical Approach." Information Systems Research 9(2): 101.

Guindon, R. (1990). "KNOWLEDGE EXPLOITED BY EXPERTS DURING SOFTWARE SYSTEM-DESIGN." International Journal of Man-Machine Studies 33(3): 279-304.

Guindon, R. and B. Curtis (1988). Control of cognitive processes during software design: what tools are needed? Proceedings of the SIGCHI conference on Human factors in computing systems. Washington, D.C., United States, ACM.

Guzman, G. A. C. and J. Wilson (2005). "The 'soft' dimension of organizational knowledge transfer." Journal of Knowledge Management 9(2): 59-74.

Hair, J. F., R. E. Anderson, et al. (1998). Multivariate data analysis. Upper Saddle River, Prentice-Hall.

Hall, T., S. Beecham, et al. (2008). The impact of staff turnover on software projects: the importance of understanding what makes software practitioners tick. Proceedings of the 2008 ACM SIGMIS CPR conference on Computer personnel doctoral consortium and research. Charlottesville, VA, USA, ACM.

Hamilton, S. and B. Ives (1982). "MIS Research Strategies." Information & Management 5: 339-347.

Hanakawa, N., K. Matsumoto, et al. (2002). "A Knowledge-Based Software Process Simulation Model." Annals of Software Engineering 14(1): 383-406.

He, J. (2004). Knowledge impacts of user participation: a cognitive perspective. Proceedings of the 2004 SIGMIS conference on Computer personnel research: Careers, culture, and ethics in a networked environment. Tucson, AZ, USA, ACM.

Henderson, J. C. and S. Lee (1992). "Managing I/S design teams: A control theories perspective." Management Science 38(6): 757.

Henninger, S. (1995). Developing domain knowledge through the reuse of project experiences. Proceedings of the 1995 Symposium on Software reusability. Seattle, Washington, United States, ACM.

Henninger, S. (1997). "Case-Based Knowledge Management Tools for Software Development." Automated Software Engineering 4(3): 319-340.

Henry, G. (1990). Practical Sampling. London, Sage Publications.

Herbsleb, J. D. (2007). Global Software Engineering: The Future of Socio-technical Coordination. 2007 Future of Software Engineering, IEEE Computer Society.

Herbsleb, J. D. and E. Kuwana (1993). Preserving knowledge in design projects: what designers need to know. Proceedings of the SIGCHI conference on Human factors in computing systems, ACM Press: 7--14.

Hewitt, B. and D. Walz (2005). Using shared leadership to foster knowledge sharing in Information Systems Development Projects. 38th International conference on system sciences, Hawaii, IEEE.

Hihn, J. and H. Habib-agahi (1991). Cost estimation of software intensive projects: a survey of current practices Proceedings of the 13th international conference on Software engineering Austin, Texas, United States IEEE Computer Society Press: 276-287

Hinds, P. and M. Mortensen (2005). "Understanding Conflict in Geographically Distributed Teams: The Moderating Effects of Shared Identity, Shared Context, and Spontaneous Communication." Organization Science 16(3): 290-307.

Huang, H. and E. M. Trauth (2007). Cultural influences and globally distributed information systems development: experiences from Chinese IT professionals. Proceedings of the 2007 ACM SIGMIS CPR conference on Computer personnel research: The global information technology workforce. St. Louis, Missouri, USA, ACM.

Humphrey, W. S. (1995). A discipline for Software Engineering, Addison Wesley Longman (Singapore) Pte Ltd, Indian Branch.

Hunter, G. (1994). "Excellent systems analysts: key audience perceptions." SIGCPR Comput. Pers. 15(1): 15-31.

Hunter, G. and S. Palvia (1996). "Ideal, advertised and actual systems analyst skills: the Singapore context." Information Technology & People 9(1).

IEEE. (2004, 20 Nov 2007). "Guide to the Software Engineering Body of Knowledge." from **http://www.swebok.org/htmlformat.html**.

Igbaria, M. and L. Chidambaram (1997). "The impact of gender on career success of information systems professionals: A human-capital perspective." Information Technology & People 10(1).

Iscoe, N., G. B. Williams, et al. (1991). Domain modeling for software engineering. Proceedings of the 13th international conference on Software engineering. Austin, Texas, United States, IEEE Computer Society Press.

ISO-SPICE. (2008). "What is SPICE?" Retrieved 20th May 2008, 2008, from **http://www.sqi.gu.edu.au/spice/what.html**.

Ives, B. and M. H. Olson (1981). "Manager or Technician? The Nature of the Information Systems Manager's Job." MIS Quarterly 5(4): 49.

Jacobson, I. (2009). Much ado about next to nothing. Information Age. Sydney, Australian Computer Society. June/July 2009.

Jain, A. and B. Boehm (2005). Developing a theory of value-based software engineering. Proceedings of the seventh international workshop on Economics-driven software engineering research. St. Louis, Missouri, ACM.

Janz, T. and P. Licker (1986). Transforming a measure of corporate culture to the information systems area: the IS review. Proceedings of the twenty-second annual computer personnel research conference on Computer personnel research conference. Calgary, Canada, ACM.

Jeffery, R. (2006). Software engineering theory and inter-disciplinary research. Proceedings of the 2006 international workshop on Workshop on interdisciplinary software engineering research. Shanghai, China, ACM.

Jiang, J. J., G. Klein, et al. (1999). "The missing link between systems analysts' actions and skills." Information Systems Journal 9(1): 21-33.

Joshi, K. D. and S. Sarker (2006). Examining the Role of Knowledge, Source, Recipient, Relational, and Situational Context on Knowledge Transfer Among Face-to-Face ISD Teams. Proceedings of the 39th Hawaii International Conference on System Sciences - 2006, Hawaii, IEEE.

Kannan, G. and K. B. Akhilesh (2002). "Human capital knowledge value added : A case study in infotech." Journal of Intellectual Capital 3(2): 167-179.

Kantrowitz, T. M. (2005). Ph.D Thesis, Development and Construct Validation of a Measure of Soft Skills Performance. Psychology, Georgia Institute of Technology.

Kaplan, B. and D. Duchon (1988). "Combining Qualitative and Quantitative Methods in Information Systems Research: A Case Study." MIS Quarterly 12(4): 570.

Klappholz, D., L. Bernstien, et al. (2003). Assessing Attitude towards, Knowledge of and Ability to apply, Software development process. Proceedings of the 16th Conference on Software Engineering Education and Training, IEEE Computer Society.

Klaus-Dieter, A., F. Bomarius, et al. (2000). "Knowledge Management for building Learning Software Organisations." Information Systems Frontiers 2(3/4): 349-367.

Klein, G., J. J. Jiang, et al. (1999). "Skill coverage in project teams." The Journal of Computer Information Systems 40(1): 69-75.

Kraemer, K. L. and W. H. Dutton, Eds. (1991). Survey research in the study of management information system. The information systems research challenge - survey research methods. Boston, Harvard Business School Research Colloquium.

Krebs, V. (2008). "Social Network Analysis, A Brief Introduction." Retrieved 1 Sep 2008, 2008, from **http://www.orgnet.com/inflow3.html**.

Kulpa, M. (2007, 20 Nov 2007). "Why Should I Use the People CMM?" Retrieved Nov-2007, from **http://www.stsc.hill.af.mil/crosstalk/2007/11/0711Kulpa.html**.

Land, L. P. W., A. Aurum, et al. (2001). Capturing Implicit Software Engineering Knowledge. Australian Software Engineering Conference. Canberra, Australia, IEEE Computer Society.

LaToza, T. D., G. Venolia, et al. (2006). Maintaining mental models: a study of developer work habits. ICSE '06: Proceeding of the 28th international conference on Software engineering, New York, NY, USA, ACM Press.

Lave, J. and E. Wenger (1991). Situated learning : legitimate peripheral participation. Cambridge [England] ; New York, Cambridge University Press.

Lee, A. S. and R. L. Baskerville (2003). "Generalizing Generalizability in Information Systems Research." Information Systems Research 14(3): 221.

Lee, D. M. S., E. M. Trauth, et al. (1995). "Critical skills and knowledge requirements of IS professionals: A joint academic/industry." MIS Quarterly 19(3): 313.

Lee, J.-Y. and C. Reigeluth (2003). "Formative Research on the Heuristic Task Analysis Process." ETR&D 51(4).

Leitheiser, R. L. (1992). "MIS Skills for the 1990s: A Survey of MIS Managers' Perceptions." Journal of Management Information Systems 9(1): 69.

Lemmen, K., F. Mulder, et al. (1999). "An empirical study on the educational effects of a course in Method Engineering for Information systems." Education and Information Technologies 4(2): 181-202.

Leon, M. V. S. (2002). "Intellectual capital : Managerial perceptions of organisational knowledge resources." Journal of Intellectual Capital 3(2): 149-166.

Lerouge, C., S. Newton, et al. (2005). "Exploring the systems analyst skill set: Perceptions, preferences, age, and gender." Journal Of Computer Information Systems 45(3): 12-23.

Lesser, E. L., Ed. (2000). Leveraging social capital in organizations. Knowledge and social capital. Woburn, MA, USA, Butterworth-Heinemann.

Lethbridge, T. (2000). "What knowledge is important to a software professional." IEEE Computer: 44-50.

Lethbridge, T., S. E. Sim, et al. (2005). "Studying Software Engineers: Data Collection Techniques for Software Field Studies." Empirical Softw. Engg. 10(3): 311-341.

Lindgren, R., O. Henfridsson, et al. (2004). "Design principles for competence management systems: A synthesis of an action research study." MIS Quarterly 28(3): 435.

Litecky, C. R., K. P. Arnett, et al. (2004). "The paradox of soft skills versus technical skills in IS hiring." Journal Of Computer Information Systems 45(1): 69-76.

Liu, X., L. C. Liu, et al. (2003). "An Examination of Job Skills Posted on Internet Databases Implications for Information Systems Degree Programs." Journal of Education for Business 78(4): 191.

Lytras, M. and A. Pouloudi (2003). "Project management as a knowledge management primer: the learning infrastructure in knowledge-intensive organizations: projects as knowledge transformations and beyond." The Learning Organization 10(4).

Macquarie Online Dictionary (2008). Macquarie Online Dictionary.

Madsen, S., K. Kautz, et al. (2006). "A framework for understanding how a unique and local IS development method emerges in practice." Eur. J. Inf. Syst. 15(2): 225-238.

Mandl, H., A. Frank, et al. (2008). "Organizational Culture as Tacit Knowledge in Action : a case study of Austrian Bioenergy Centre Wieselburg." Retrieved 15 Jan 2009, 2009, from **http://www.sol-austria.at/contents/20080115163946981M10.pdf**.

Marczak, S., D. Damian, et al. (2008). Information brokers in requirement-dependency social networks. 16th IEEE International requirements engineering conference: 53-60.

Mason, L. (2007). "Introduction: Bridging the Cognitive and Sociocultural Approaches in Research on Conceptual Change: Is it Feasible?" Educational Psychologist 42(1): 1-7.

Mathiassen, L. and P. Pourkomeylian (2003). "Managing knowledge in a software organization." Journal of Knowledge Management 7(2): 63-80.

Mathiassen, L. and L. Vogelsang (2005a). "Managing knowledge in software method adoption." Int. J. Bus. Inf. Syst. 1(1/2): 102-117.

Mathiassen, L. and L. Vogelsang (2005b). The Role of Networks and Networking in Bringing Software Methods to Practice. Proceedings of the Proceedings of the 38th Annual Hawaii International Conference on System Sciences - Volume 08, IEEE Computer Society.

Mayer, R. (1992). Thinking, Problem solving, Cognition. New York, W.H. Freeman and Company.

McCubbray, D. J. (1988). The systems analyst of the 1990's. Proceedings of the ACM SIGCPR conference on Management of information systems personnel. College park, Maryland, United States, ACM Press.

McCubbray, D. J. and R. Scudder (1988). The systems analyst of the 1990's. Proceedings of the ACM SIGCPR conference on Management of information systems personnel. College park, Maryland, United States, ACM Press.

McDonald, C. (2002). "Information Systems Foundations - Karl Popper's Third World." The Australian Journal of Information Systems 9(Special edition).

McDonald, D. W. (2001). Evaluating expertise recommendations. Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work, ACM Press: 214--223.

McGaw, D. and G. Watson (1976). Political and Social Inquiry, John Wiley and Sons Inc.

Medlin, D., B. Ross, et al. (2004). Cognitive Psychology, Wiley.

Mills, J. (1985). "A pragmatic view of the system architect." Commun. ACM 28(7): 708-717.

Misic, M. (1996). "The skills needed by today's systems analysts."." Journal of Systems Management 47(3 (May-June 1996)): 34(37).

Monin, D. J. and P. Dewe (1994). Skills in an environment of turbulence: a survey of information systems professionals in New Zealand. Proceedings of the 1994 computer personnel research conference on Reinventing IS : managing information technology in changing organizations: managing information technology in changing organizations. Alexandria, Virginia, United States, ACM Press.

Mookerjee, R. (2005). "Maintaining enterprise software applications." Commun. ACM 48(11): 75-79.

Moore, Ed. (1985). A formal theory of knowledge and action. Formal Theories of the Commonsense World, Ablex Publishing.

Nachmias, D. and C. Nachmias (1976). Research methods in the social sciences. London, Edward Arnold.

Nakayama, M. and N. Sutcliffe (2001). IT skills portfolio research in SIGCPR proceedings: analysis, synthesis and proposals. Proceedings of the 2001 ACM SIGCPR conference on Computer personnel research. San Diego, California, United States, ACM Press.

Nansi, S. (1998). Ph.D Thesis, Critical information systems management issues frameworks and relationships with individual IS executives and organisational environments: A study in Singapore. International Graduate School of Management. Adelaide, University of South Australia.

Nansi, S. and D. Bennett (1998). "Requisite IS management knowledge and skills construct: a survey." SIGCPR Comput. Pers. 19(1): 3--19.

Nelson, R. R. (1991). "Educational Needs as Perceived by IS and End-User Personnel: A Survey of Knowledge and Skill Requirements." MIS Quarterly 15(4): 503.

Neuman, W. L. (2006). Social research methods : qualitative and quantitative approaches. Boston, Pearson/AandB.

Ng, T. W. H. and D. C. Feldman (2008). "The Relationship of Age to Ten Dimensions of Job Performance." Journal of Applied Psychology 93(2): 392-423.

Nord, D. and J. H. Nord (1997). "Information systems project development: knowledge and domain requirements for the systems analyst." Industrial Management & Data Systems 1: 17-24.

Nunnaly, J. (1978). Psychometric theory. New York, McGraw-Hill.

Olson, J. (2007). "Culture Surprises in Remote Software Development Teams." Retrieved 15 Nov 2007, 2007, from **http://www.acmqueue.com/modules.php?name=Content&pa=showpage&pid=105&page=3**.

Orsted, M. (2000). Software development engineer in Microsoft: a subjective view of soft skills required. Proceedings of the 22nd international conference on Software engineering. Limerick, Ireland, ACM Press.

Osterweil, L. J. (2007). A Future for Software Engineering? 2007 Future of Software Engineering, IEEE Computer Society.

Packer, M. J. and J. Goicoechea (2000). "Sociocultural and Constructivist Theories of Learning: Ontology, Not Just Epistemology." Educational Psychologist 35(4): 227-241.

Panjer, L. D., D. Damian, et al. (2008). Cooperation and coordination concerns in a distributed software development project. Proceedings of the 2008 international workshop on Cooperative and human aspects of software engineering. Leipzig, Germany, ACM.

Parsons, J. and C. Saunders (2004). "Cognitive Heuristics in Software Engineering: Applying and Extending Anchoring and Adjustment to Artefact Reuse." IEEE Transactions on Software Engineering 30(12): 873-888.

Patton, R. (2005). Software Testing, SAMS Publishing.

Pennington, N., A. Y. Lee, et al. (1995). "Cognitive activities and levels of abstraction in procedural and object-oriented design." Hum.-Comput. Interact. 10(2): 171-226.

Perkins, D. N., E. Jay, et al. (1993). "Beyond abilities - A dispositional theory of thinking." Merrill-Palmer Quarterly-Journal of Developmental Psychology 39(1): 1-21.

Perry, D. E., H. Siy, et al. (2001). "Parallel changes in large-scale software development: an observational case study." ACM Trans. Softw. Eng. Methodol. 10(3): 308-337.

Petter, S., M. Lars, et al. (2007). Five keys to project knowledge sharing. IT Pro, IEEE Computer society**:** 42-46.

Philp, B. and B. Garner (2001). Knowledge Mediation in Software Quality Engineering. Australian Software Engineering Conference. Canberra, Australia, IEEE Computer Society.

Popper, K. R. (1972). Objective Knowledge : An Evolutionary Apprach. London, Clarendon Press.

Prabhakar, B., C. Litecky, et al. (2005). "IT skills in a tough job market." Commun. ACM 48(10): 91-94.

Premkumar, G. (1989). "A cognitive study of the decision-making process in a business context - implications for design of expert systems." International Journal of Man-Machine Studies 31(5): 557-572.

Pressman, R. S. (2009). Software engineering : a practitioner's approach. New York, McGraw-Hill Higher Education.

Princeton University. (2007, 15 Nov 2007). "Performance. (n.d)." from **http://dictionary.reference.com/browse/performance**.

Princeton University. (2008). "Data and Statistical Services : Interpreting regression output." Retrieved 31st December 2008, 2008, from **http://dss.princeton.edu/online_help/analysis/interpreting_regression.htm**.

Project Management Institute. (2009). "A Guide to the Project Management Body of Knowledge (PMBOK® Guide)." Retrieved 1st July 2009, 2009, from **http://www.pmi.org/Resources/Pages/Library-of-PMI-Global-Standards-projects.aspx**.

Rajeswari, K. S. and R. N. Anantharaman (2003). Development of an instrument to measure stress among software professionals: factor analytic study. Proceedings of the 2003 SIGMIS conference on Computer personnel research: Freedom in Philadelphia--leveraging differences and diversity in the IT workforce. Philadelphia, Pennsylvania, ACM.

Rasch, R. H. and H. L. Tosi (1992). "Factors affecting software developers' performance: An integrated approach." MIS Quarterly 16(3): 395.

Reedman, P. (2004). "Business analyst and Systems Analyst : they are different." Retrieved 25 Nov 2004, 2004, from **http://blogs.ittoolbox.com/eai/development/archives/business-analyst-and-systems-analyst-they-are-different-677**.

Riel, A. (1996). Object-Oriented Design Heuristics. Boston, Addison Wesley.

Rilling, J. (2002). Maximizing functional cohesion of comprehension environments by integrating user and task knowledge. Proceedings of the eighth working conference on reverse engineering, IEEE.

Robbins, S., B. Millet, et al. (2001). Organisational behaviour. Frenchs Forest, NSW, Prentice Hall.

Roberts, H. and S. Gregor (2005). "Why Australian Merchants Aren't Adopting E-Money." Journal of Research and Practice in Information Technology 37(1): 11-25.

Robey, D. and M. Newman (1996). "Sequential patterns in information systems development: an application of a social process model." ACM Trans. Inf. Syst. 14(1): 30-63.

Robillard, P. N. (1999). "The role of knowledge in software development." Commun. ACM 42(1): 87-92.

Robillard, P. N., P. d'Astous, et al. (1998). Measuring cognitive activities in software engineering. Proceedings of the 20th international conference on Software engineering. Kyoto, Japan, IEEE Computer Society.

Rodriguez, O., A. Martinez, et al. (2004). Identifying KM needs in software maintenance groups: A qualitative approach. Proceedings of the Fifth Mexican International Conference in Computer Science (ENC'04), Mexico, IEEE.

Rogoff, B. and J. Lave (1984). Everyday cognition : its development in social context. Cambridge, Mass., Harvard University Press.

Rosson, M. B. (1996). "Human factors in programming and software development." ACM Comput. Surv. 28(1): 193-195.

Rost, J. (2005). "Software Engineering Theory in Practice." IEEE Software 22(2): 96-95.

Rumizen, M. C. (2002). The complete idiot's guide to Knowledge Management. Indianapois, IN, Pearson Education.

Rummel, R. J. (1970). Applied Factor Analysis. Evanston, Northwestern University Press.

Rus, I. and M. Lindvall (2002). "Knowledge Management in Software Engineering." IEEE Software May/June(May-June-2002): 26-37.

Rus, I., M. Lindvall, et al. (2001). Knowledge Management in Software Engineering: A state-of-the-art report. Rome, NY, Fraunhofer Center for Experimental Software Engineering, Maryland and The University of Maryland.

Saeki, M. (1995). Communication, collaboration and cooperation in software development - How should we support group work in software development ? Asia Pacific Software Engineering Conference '95, IEEE.

Sage, A. and W. Rouse (1999). "Information systems frontiers in Knowledge Management." Information Systems Frontiers 1(3).

Samovar, L. A. and R. Porter (1995). Communication between cultures, Wadsworth Publishing.

Santos, G., M. Montoni, et al. (2007). Implementing SPI initiatives in small and medium-size enterprises in Brazil. Sixth International conference on the quality of ICT, IEEE.

Sapsford, R. (2007). Survey Research. London, Sage Publications.

Sarker, S. and A. S. Lee (2003). "Using a case study to test the role of three key social enablers in ERP implementation." Information & Management 40(8): 813.

Sawyer, S., K. Eschenfelder, et al. (1998). "Corporate IT skill needs: a case study of BigCo." SIGCPR Comput. Pers. 19(2): 27-41.

Sawyer, S., J. Farber, et al. (1996). "Supporting the social processes of software development." Information Technology & People 10(1): 46-62.

Sawyer, S. and P. J. Guinan (1998). "Software development: Processes and Performance." IBM Systems journal 37(4).

Schein, E. (2004). Organisational culture and leadership, John Wiley and Sons.

Scheuren, F. (2007). "What is a survey." Retrieved 15 August 2007, 2007, from **http://www.whatisasurvey.info/**.

Schneider, K. and J.-P. v. Hunnius (2003). Effective experience repositories for software engineering. Proceedings of the 25th International Conference on Software Engineering. Portland, Oregon, IEEE Computer Society.

Schwab, D. (2005). Research Methods for Organizational Studies: Second Edition, Routledge.

Schwalbe, K. (2002). Information Technology Project Management. Massachusetts, Course Technology Thomason Learning.

Sekaran, U. (2003). Research methods for business: A skill building approach, John Wiley and Sons Inc.

Selamat, M. H. and J. Choudrie (2004). "The diffusion of tacit knowledge and its implications on information systems: The role of meta-abilities." Journal of Knowledge Management 8(2): 128-139.

Sen, A. (1997). "The role of opportunism in the software design reuse process." IEEE Transactions on Software Engineering 23(7): 418-436.

SFIA. (2008). "Skills Framework for the Information Age." Retrieved 20 Sep 2008, 2008, from **http://www.sfia.org.uk/**.

Shariq, S. Z. (1999). "How does knowledge transform as it is transferred? Speculations on the possibility of a cognitive theory of knowledgescapes." Journal of Knowledge Management 3(4): 243-251.

Sheng, Y., J. M. Pearson, et al. (2003). "Organizational Culture and Employees' Computer Self-Efficacy: An Empirical Study." Information Resources Management Journal 16(3): 42-48.

Shi, Z. (2007). "An empirical test of IS leadership as the driving force in improving IS performance." Journal of Information, Information Technology and Organizations 2.

Simerson, B. K. and M. L. Venn (2007). The manager as leader, Greenwood Publishing Group.

Smith, S. L. and J. N. Mosier. (1986). "Guidelines for designing user interface software." Retrieved 15 March 2009, 2009, from **http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.84.8930&rep=rep1&type=pdf**.

Software Engineering Institute. (2004). "Process Maturity Profile : CMMI v.1.1 Class A Appraisal Results." Retrieved 09 Sep 2004, 2004, from **http://www.sei.cmu.edu/sema/pdf/CMMI/2004aug.pdf**.

Sommerville, I. (2001). Software Engineering. Essex, England, Addison-Wesley.

Stair, R. and G. Reynolds (2008). Principles of Information systems USA, Course Technology.

Stanek, D. M. (1993). "Modelling Perceptions and Preference of Home-based and Center-based Telecommuting." Retrieved 15 Aug 2007, 2007, from **http://www.its.ucdavis.edu/telecom/r11/factan.html**.

Steen, O. (2007). "Practical knowledge and its importance for software product quality." Inf. Softw. Technol. 49(6): 625-636.

Sternberg, R. J. (1989). "Domain-generality versus domain-specificity: The life and impending death of a false dichotomy." Merrill-Palmer Quarterly 35(1): 115-130.

Sternberg, R. J. and T. Ben-Zeev (2001). Complex Cognition: The psychology of human thought. Newyork, Oxford University Press.

Suan, H., Bok, et al. (2000). Software engineering productivity measurement using function points: a case study. 15**:** pp.79.

Sulaiman, S., N. A. Rashid, et al. (208). Novices' Expectations in Gaining Theoretical Knowledge and Practical Skills in Software Development or Maintenance. Software Engineering Conference. Innsbruck, Austria, www.actapress.com.

Summers, T. P. and S. B. Summers (1997). "Strategic Skills Analysis for Selection and Development." Human Resource Planning 20(3): 14-19.

Sveiby, K. (1997). The new organizational wealth : managing and measuring knowledge-based assets. San Francisco, Berrett-Koehler Publishers.

Tabachnick, B. and L. Fidell (1989). Using Multivariate Statistics. Newyork, HarperCollins Publishers.

Tanner, K., Ed. (2000). Survey research. Research methods for students and professionals : Information management and systems. Wagga Wagga, Centre for information studies, Charles Sturt University, New South Wales.

Teijlingen, E. R. v. and V. Hundley. (2001). "The importance of pilot studies." Retrieved 15th Jan 2006, 2006, from **http://sru.soc.surrey.ac.uk/SRU35.html**.

Telliglu, H. and I. Wagner (1999). "Software Cultures : Exploring cultural practices in managing heterogeneity within system design." Commun. ACM 42(12): 71-77.

Teo, T. and V. K. G. Lim (1996). "Factors influencing personal computer usage: the gender gap." Women in Management Review 11(8).

Thawyer, R. (2004). "Software engineering glossary : Software configuration management domain." Retrieved 09 September 2004, from **http://csdl.computer.org/comp/mags/so/2003/04/s40c3.pdf**.

The Open group. (2007a). "TOGAF-V8-M1-Management-Overview." Retrieved 20th Sep 2007, 2007, from **http://www.togaf.org and then click on TOGAF-V8-M1-Management-Overview.pdf**.

The Open group. (2007b). "TOGAF Architecture Skills Framework." from **http://www.opengroup.org/architecture/togaf8-doc/arch/chap30.html**.

Tiwana, A. (2000). The Knowledge Management Toolkit : Practical techniques for building a knowledge management system. Upper Saddle River, N.J., Pearson Education Asia Pte Ltd.

Todd, P. A., J. D. McKeen, et al. (1995). "The evolution of IS job skills: A content analysis of IS job advertisements from 1970 to 1990." MIS Quarterly 19(1): 1.

Tondl, L. (2007). "Rational actions and the integration of knowledge." Journal for General Philosophy of science 38: 91-110.

Trochim, W. (2006). "Research Methods Knowledge Base." Retrieved 1 Jan 2008, 2008, from **http://www.socialresearchmethods.net/kb/variable.php**.

Tye, E., R. S. K. Poon, et al. (1995). "Information systems skills: Achieving alignment between the curriculum and the needs of the IS professionals in the future." Data Base For Advances In Information Systems 26(4): 47-61.

Valiris, G. and L. Laios (1995). "DESIGN AND VALIDATION OF KNOWLEDGE ACQUISITION TOOLS IN A BUSINESS DOMAIN." Behaviour & Information Technology 14(2): 121-131.

Vitalari, N. and G. Dickson (1983). "Problem solving for effective systems analysis: an experimental exploration." Commun. ACM 26(11): 948-956.

Vitalari, N. P. (1985). "Knowledge as a Basis for Expertise in Systems Analysis: An Empirical Study." MIS Quarterly 9(3): 221.

Wade, M. R. and M. Parent (2001). "Relationships Between Job Skills and Performance: A Study of Webmasters." Journal of Management Information Systems 18(3): 71.

Wagner, W. P., M. K. Najdawi, et al. (2001). "Selection of knowledge acquisition techniques based upon the problem domain characteristics of production and operations management expert systems." Expert Systems 18(2): 76-87.

Walz, D. B., J. Elam, et al. (1993). "Inside a software design team: knowledge acquisition, sharing, and integration." Commun. ACM 36(10): 63--77.

Ward, J. and A. Aurum (2004). Knowledge Management in Software Engineering - Describing the Process. Proceedings of the 2004 Australian Software Engineering Conference, IEEE Computer Society.

Wasserman, A. (1996). "Toward a discipline of software engineering." IEEE Software November(23-27).

Watson, H., D. Young, et al. (1990a). "Requisite skills for new MIS hires." SIGMIS Database 21(1): 20-29.

Watson, H. J., D. Young, et al. (1990b). "Requisite skills for new MIS hires." SIGMIS Database 21(1): 20-29.

Wenger, E. (1998). Communities of Practice. Cambridge, Cambridge University Press.

Wertsch, J. V. and B. Rogoff (1984). Children's learning in the "zone of proximal development". San Francisco, Jossey-Bass.

White, K. B. and R. Leifer (1986). "Information Systems Development Success: Perspectives from Project Team Participants." MIS Quarterly 10: 215-223.

Wikipedia contributors. (2007a). "Configuration." Retrieved 20 Aug 2007, 2007, from **http://en.wikipedia.org/wiki/Configuration**.

Wikipedia contributors. (2007b). "Programmer." Retrieved 20 Aug 2007, 2007, from **http://en.wikipedia.org/wiki/Programmer**.

Williamson, K., F. Burstein, et al., Eds. (2000). Introduction to research in relation to professional practice. Research methods for students and professionals : Information management and systems. Wagga Wagga, Centre for information studies, Charles Sturt University, New South Wales.

Wingreen, S. C., J. E. Blanton, et al. (2003). Toward a social cognitive basis for the IT professional's person-organization fit research-in-progress. Proceedings of the 2003 SIGMIS conference on Computer personnel research: Freedom in Philadelphia--leveraging differences and diversity in the IT workforce. Philadelphia, Pennsylvania, ACM Press.

Wolf, T., A. Schroter, et al. (2009). "Mining Task-based social networks to explore collaboration in software teams." IEEE Software(January/February 2009): 58-66.

Wooten, W. (1993). "Using Knowledge, Skill and Ability (KSA) Data to Identify Career Pathing Opportunities - an Application of Job Analysis to Internal Manpower-Planning." Public Personnel Management 22(4): 551-563.

WordNet 2.0 Search. (2004). "Overview of software." Retrieved 9th Sep 2004, 2004, from **http://www.cogsci.princeton.edu/cgi-bin/webwn?stage=1&word=software**.

Yamauchi, Y., J. Whalen, et al. (2006). The problem of knowledge decoupling in software development projects. Proceedings of the 28th international conference on Software engineering. Shanghai, China, ACM.

Yaobin, L. and Z. Tao (2007). "A research of consumer's initial trust in online stores in China." Journal of Research and Practice in Information Technology 39(2): 167-180.

Ye, Y. (2006). Supporting software development as knowledge-intensive and collaborative activity. WISER '06: Proceedings of the 2006 international workshop on Workshop on interdisciplinary software engineering research, New York, NY, USA, ACM Press.

Yin, R. (2003). Case study research Design and Methods. London, Sage Publications.

Zannier, C. and F. Maurer (2007). Social Factors Relevant to Capturing Design Decisions. Proceedings of the Second Workshop on Sharing and Reusing architectural Knowledge Architecture, Rationale, and Design Intent, IEEE Computer Society.

Zmud, R. (1983). Information Systems in Organizations. Oakland, New Jersey, Scott, Foresman and Company.

Zmud, R. W. (1979). "Individual Differences and MIS Success: A review of the empirical literature." Management Science 25(10): 966.

10 Glossary and Acronyms

Terminology/Acronym	Description
3GL	Third Generation Languages such as COBOL, C#.NET
ABS	Australian Bureau of Statistics
ACM	Association for Computing Machinery
ACS	Australian Computer Society
ACT	Adaptive Control Theory: A model to look at the way human mind operates based its cognitive nature. The theory provides explanations on knowledge and action.
Anderson's ACT Theory	Anderson's Adaptive Control of Thought Theory.
ANOVA	Analysis of Variance : A statistical procedure to detect relationships between variables in the data
Application system knowledge	Technical and functional knowledge of the application system. Referred to as configuration knowledge in this thesis
Articulable tacit knowledge	Tacit knowledge that is articulable or codifiable (Busch et al., 2001).
ASP.NET	Active Serve Pages.NET – A web programming language from Microsoft Corporation
BA	Business Analyst
Billett's Model	Billett's(1995) Model of Knowledge and Performance
Billett's Model of Knowledge and Performance	Billett (1995) reconciled theories of social and cognitive origins of knowledge and came up with a model of knowledge and action for an expert. He theorised and provided evidence that in a problem situation both prior cognitive knowledge as well as social circumstantial knowledge are required to perform action.
Business domain area knowledge	Knowledge of the business domain area in which the application software system operates. Examples: insurance and banking

Terminology/Acronym	Description
CMMI	Capability Maturity Model Integrated: A process maturity model proposed by Software Engineering Institute (SEI).
CMS	The name of the application system in the case study project.
COBOL	A programming language mainly used for business applications
Cognitive origins of knowledge	Theories of knowledge and action predominantly based on cognitive nature of human mind.
Combination roles	Multiple roles performed by an individual at the same time. Example : A person performing both project manager and business analyst role.
Configuration knowledge	The technical and functional knowledge of the application software system.
Content validity	Content validity, also referred to as construct validity, relates to the representativeness of sampling adequacy of the questionnaire regarding the content or the theoretical constructs to be measured (Burns cited in Cavana, 2001 p.238).
Contextual knowledge	Knowledge of the contextual environment in which an individual performs the role. For example, the cultural knowledge of the organisation is a contextual knowledge that is required to perform a specific role. Context emerges through the individual interpretation of a situation including others and artefacts (Augier et al., 2001). Also referred to as Circumstantial knowledge, Social circumstantial knowledge, Knowledge of social origins, Local knowledge, Organisational specific knowledge, Circumstantial knowledge.
COOLGEN	An enterprise Computer Aided SE (CASE) tool for application development
Core Technique skills	The knowledge, skills and abilities that are directly relevant to performing a role. Example : Business analysts require business analysis skills. Programmers require coding skills.
Cronbach's alpha	A statistical value to ascertain reliability of factors
CSS	Cascade Style Sheet
Cultural knowledge	The cultural knowledge of the organisation.

Terminology/Acronym	Description
DB2	IBM Corp's RDBMS product named DB2
DBA	Data Base Administrator
DCC	Data Collection Contact. Individuals who helped with collection of responses for the general-survey.
Declarative knowledge	Declarative knowledge comprises of encoded facts, information, assertions, concepts and propositions (Anderson, 1982; Anderson, 1993).
Domain	Particular topic (e.g. knowledge of car engines or of the solar system would be distinct domains of knowledge)
Domain specific knowledge	a relative self-contained collection of knowledge about a particular topic (e.g. knowledge of car engines or of the solar system would be distinct domains of knowledge) (Eysenck and Keane, 2005 p.455)
DSS	Decision Support Systems
easySAP	An Enterprise Resource Planning product from SAP Corporation, Germany.
EFA	Exploratory Factor Analysis : A statistical method to explore underlying factor structure in data
EIS	Executive Information Systems
Enterprise System	Refers to Enterprise Resource Planning (ERP) type systems in Chan's (2003) research.
Explicit knowledge	Explicit knowledge is knowledge that can be codified and transmitted in various forms like e-mail, documents or databases (Tiwana, 2000 p.66).
Face validity	Face validity addresses the concern of whether the questionnaire appears to measure the concepts being investigated (Burns cited in Cavana, 2001 p.238)
Factor Analysis	a statistical technique for determining which among a set of variables, for which statistics are held, exert the greatest influence on the matter being investigated (Macquarie Online Dictionary, 2008)

Terminology/Acronym	Description
General systems knowledge	Same as Systems knowledge
General-survey	The survey of participants working SE teams from different cities across the world conducted in this research
Heuristics knowledge	Knowledge of short cuts, rules of thumb and other methods by which an expert is able to solve complex problems.
HRM	Human Resource Management
HTA	Heuristic Task Analysis
HTML CSS	Hyper Text Mark-up Language
HW/SW	Hardware/Software
ibid.	In the same book, chapter, page etc (Macquarie Online Dictionary, 2008)
IBM	International Business Machines Corp
IBM-DB2	IBM Corp's RDBMS product called DB2
ICT	Information and Communications Technology
IEEE	Institute of Electrical and Electronic Engineers
Inarticulable tacit knowledge	For some types of tacit knowledge, individuals may not even know that they know that knowledge is held by them and that they use it in performing their job. It is difficult to understand, extract and codify such knowledge. These are referred to as inarticulable tacit knowledge (Busch et al., 2001).
IRS	Inland Revenue Service
IS	Information Systems
ISD	Information Systems Design
ISO	International Standards Organisation
ISO-SPICE	International Standards Organisation – Software Process Improvement and Capability dEtermination (ISO-SPICE, 2008).
ISPF	Interactive System Productivity Facility : Used to access OS/390 Mainframe services interactively through UI.
IT	Information Technology

Terminology/Acronym	Description
Item	Operational variable used in research; also referred to as operational item.
ITPC	Information Technology Professional Competence : Defined as the degree to which one possesses expert knowledge and skills, and exhibits behaviours that are effective in performing the design, development and consulting duties relevant to the contemporary Information Technology (IT) profession (Blanton et al., 1998).
KM	Knowledge Management
KMO	Kaiser-Meyer-Olkin (KMO) Measure of Sampling Adequacy. The KMO should be above 0.6 for factorability (Hair et al., 1998)
Knowledge Index	An index used to measure a specific type of knowledge of an individual in this thesis (see section 11.1 for more details)
Knowledge Management	Knowledge Management is defined as the process by which an organisation generates wealth from its individuals and knowledge-based assets (Bukowitz and Williams, 1999).
KSA	Knowledge, Skills and Abilities
Microsoft Excel	Microsoft corporation’s spreadsheet product
MIS	Management Information Systems
NTT, Japan	National Telephone and Telegraph company of Japan.
Op.cit.	In the work cited (Macquarie Online Dictionary, 2008)
Operational item	Operational measures by which scores are obtained from the cases studied (Schwab, 2005). Also referred to as operational variable.
Operational knowledge	Knowledge required to perform well in a given position (Beazley et al., 2002)
Operational variable	Operational measures by which scores are obtained from the cases studied (Schwab, 2005). Also referred to as operational item.

Terminology/Acronym	Description
Other roles	Roles not directly considered by this research. i.e. roles other than the following : Business analyst, tester, designer, designer, project manager, programmer, programmer-analyst and architect.
PA	Programmer Analyst
PAF	Principal Axis Factoring – A statistical method within EFA.
PCMM	People Capability Maturity Model published by SE Institute (see 11.14 for more details)
PDI	Power Distance Index
Performance	The "act of performing; of doing something successfully; using knowledge as distinguished from merely possessing it" (Princeton University, 2007).
Perl	A programming language
PM	Project Manager
PMBOK	Project Management Body Of Knowledge published by Project Management Institute
Procedural knowledge	Procedural knowledge comprises of encoded techniques, skills and the ability to secure goals (Anderson, 1982; Eysenck and Keane, 2005) and includes problem solving skills (Anderson, 1993). Procedural knowledge is encoded in terms of production rules that are condition-action pairs (Anderson, 1993).
Procedural memory	Same as procedural knowledge
Process knowledge	Knowledge of the processes and procedures used in the SE team and organisation while delivering the expected outcomes.
Production memory	Same as procedural knowledge
RACF	Resource Access Control Facility : An access control facility for the OS/390 based mainframe.
Radial architecture	A layered architecture model separating channel delivery, business functionality and technical layers.
RDBMS	Relational Data Base Management System
Role	The role played by an individual in a SE team.

Terminology/Acronym	Description
Scale	A scale is a mechanism by which individual responses are distinguished as to how they differ from one another on the variables that are of interest to the research study (Sekaran, 2003).
SDLC	Software Development Life Cycle
SE	Software Engineering
SEI	Software Engineering Institute
SNA	Social Network Analysis
Social circumstantial knowledge	Knowledge of the contextual environment in which an individual performs the role. For example, the cultural knowledge of the organisation is a contextual knowledge that is required to perform a specific role. Also referred to as Circumstantial knowledge, Knowledge of social origins, Local knowledge, Organisational specific knowledge, Circumstantial knowledge, Contextual knowledge in the thesis.
SNA	Social Network Analysis
Social network knowledge	An understanding of the crucial relationships that are required to perform the job (Beazley et al., 2002).
Social origin theories of knowledge	Theories of knowledge and action predominantly based on social, cultural and historical origins.
Soft skills	the skills needed to perform jobs where job requirements are defined in terms of expected outcomes, but the processes used to achieve the outcomes may vary widely (Ehrlich, 2004)
SPICE	Same as ISO-SPICE
SPSS	A software product for statistical analysis
SQL	Structured Query Language
SWEBOK	SE Body of Knowledge published by IEEE (see 11.13 for more details)
Systems knowledge	Systems knowledge is defined as ‘an understanding of the interplay of cause-and-effect relationships that is essential for sound decision making’ (Beazley et al., 2002).

Terminology/Acronym	Description
Tacit knowledge	The knowledge that we do not know that we know is referred to as tacit knowledge (Rumizen, 2002 p.8). Tacit knowledge is personal, context-specific knowledge which is difficult to articulate (Rumizen, 2002 p.8) and is usually in the heads of people (Tiwana, 2000 p.66).
Technical skills	Bailey and Stefaniak (2001) define technical skills as the specialised knowledge and skills required to work in a software development team such as the knowledge of computer languages, operating systems, data bases.
Technique skills	Specialised knowledge and skills required to perform a specific role in a SE team.
TOGAF	The Open Group Architecture Framework (The Open group, 2007a)
UI	User Interface
UNIANOVA	Univariable ANOVA
USA	United States of America
Users	Users of an application system.
Variable	A variable is any entity that can take on different values (Trochim, 2006). The formulated theoretical framework consists of variables or factors that are operationalised into operational items or operational variables.
VB	Microsoft Visual Basic programming language
Visual Studio.NET 2003	Integrated development environment for .NET from Microsoft Corporation.
Working knowledge	Temporary memory in human mind used during cognitive processing.
Working memory	Same as working knowledge

11 Appendices

11.1 Knowledge index

Brown (2000) defines an index as a combination of several separate measures added together or averaged to result in a single, overall indicator of performance. A knowledge index of individual for a given knowledge type is calculated by using tabulating the information as shown in the following table (Table 11.1). The first column in the table lists the knowledge items used in the index calculation. The second column lists the knowledge of the individual for those knowledge items. The third optional column of weights allows one to assign a weight for different knowledge item depending on its perceived importance to performing the job. The fourth column shows the weighted knowledge rating.

Knowledge Item A_a	Knowledge/Skill rating (Scale 0 = No knowledge, 1 – Basic, 2 – Competent, 3 – Expert) (S_a)	Weight for Operational Knowledge type (W_a) (Optional)	Weighted Knowledge Points (W_a*S_a)
Item 1			
Item 2			
Item 3			
Item 4			
			$\sum W_a S_a$

Table 11.1. Knowledge Index calculation tabulation

Max Knowledge Rating (M_c) is a constant value that denotes the maximum value on the scale used. In this case since the scale 3 (expert) is the maximum value for the scale.

Knowledge Index for a specific operational knowledge type = $\sum W_a S_a / (n * M_c)$

This knowledge index is used for measuring the initial knowledge and current knowledge of an individual for identified operational knowledge types. As suggested by Neuman (2006), since there are no theoretical reasons for assigning different weights, each item is assigned equal weight. These types of indices have been used in the literature. For example, Shu Nansi (1998b) defined new indices such as IS competence index, IS assimilation index while testing the critical information systems management issues.

11.2 Template for letter of informed consent for case study research

School of Information Sciences and Engineering

University of Canberra

PARTICIPATION AGREEMENT (Consent form)

Please read the following information carefully.

In this research project, we are examining the knowledge required of developers to perform various jobs in an application software project.

We believe that this research should be conducted with people like you, that is, people who work in real software development environments.
There are two parts to this data collection process
Part 1 : A survey questionnaire. (Approximately 1 hr)
Part 2 : Interview with the researcher (Approx : 15 minutes)
The information captured will be used to

- Help determine the knowledge requirements to perform the type of job you do.
- Understand the contributions of different types of knowledge to job performance.
- Develop new indices to measure knowledge

Any information you, or other users, give will remain anonymous and will be kept in strict confidence. Your responses will not be used for any purpose other than this research. We might want to quote you as a 'participant' but we will ensure there is no identifying information.

It is important to emphasize that it is *not* an evaluation of your work performance or skills, but rather it is aimed at identifying and measuring knowledge required to perform jobs of your type. There are no 'right' or 'wrong' answers, and if you have any problems in understanding the research questions, then feel free to ask the researcher. The observations and information we collect are important and will help us make recommendations to improve project and knowledge management in application software development projects.

Your participation is entirely voluntary; you may choose to withdraw from the project at any time.

If you have read the above information carefully, understand and consent to participating in the project, please complete and sign below.

-- ----------------------------------- -------------------

(Participant's name) (Signature)

(Date)

-- ----------------------------------- --------------------

(Researcher's name) (Signature)

(Date)

11.3 Case study research questionnaire

Case study project
Knowledge and Performance: Team member interview questionnaire
Section 1

PART A: DEMOGRAPHICS AND WORK EXPERIENCE

Question A.1: What is the length of time you have been working in application software delivery projects? (In years and months) ___________

Question A.2: What is your gender? ☐ **Male** ☐ **Female**

Question A.3: What is your mode of work?

☐ Predominantly work at the organisation's location with other members of the team.
☐ Work remotely from home
☐ Any other, please specify. ___________________________________

Question A.4: What is the highest qualification you have obtained?

☐ High school certificate ☐ Vocational Certificate ☐ Diploma ☐ Bachelor's degree ☐ Master's Degree/Post Graduate ☐ Doctorate
Other (specify) _______________

Question A.5: What is the field of study for your highest qualification? (E.g. Maths, Computer science) _________________________

The following questions relate to the application software project team with which you are currently working.

Question A.6: What is your main role in the software project team?

☐ Business Analyst ☐ Programmer ☐ Architect ☐ Designer ☐ Tester ☐ Project Manager ☐ Other (Please specify) ______________________

Question A.7: How many years have you been working with this software project team? (In years & months) _________

Question A.8: What is the industry segment of the application software? (Eg. Insurance, transport, Banking, Government, etc) ___________________________

Question A.9 Indicate the relative time you have spent working on the various aspects of your current work. Circle the appropriate option.

	not applicable	**very rarely**	**rarely**	**regularly**	**very regularly**	**most of the time**
Requirements analysis	•	•	•	•	•	•
Architectural design	•	•	•	•	•	•
High level design	•	•	•	•	•	•
Low level design	•	•	•	•	•	•
Coding and unit testing	•	•	•	•	•	•
Function testing	•	•	•	•	•	•
Integration testing	•	•	•	•	•	•
Configuration management	•	•	•	•	•	•
Quality assurance	•	•	•	•	•	•
Preparation of manuals	•	•	•	•	•	•
Implementation	•	•	•	•	•	•
Production support	•	•	•	•	•	•
Project Management	•	•	•	•	•	•
Other-1 (write here)	•	•	•	•	•	•
Other-2 (write here)	•	•	•	•	•	•

Question A.12: Have you heard or used the term 'Knowledge Management' during communication with others in your software project team?

☐ Yes ☐ No

Question A.13: Does your organisation adhere to any type of software process model or standard?

- ☐ Capability Maturity Model Integrated (CMMI)
- ☐ Software Process Improvement and Capability dEtermination (SPICE)
- ☐ International Standards Organisation (ISO-9001)
- ☐ Rational Unified process (RUP)
- ☐ PRINCE2
- ☐ Any other , please specify. ____

Question A.14: If your organisation adheres to any process model/standard, write down the formal accreditation levels on process maturity

Question A.15: What is your opinion on software development processes in your software project team?

- ☐ Initial: Processes are usually ad-hoc or chaotic
- ☐ Managed : Projects have ensured that requirements are managed and that processes are planned, performed, measured, and controlled
- ☐ Defined: Processes are well characterized and understood, and are described in standards, procedures, tools, and methods
- ☐ Quantitatively Managed : Metrics collected on the managed processes and controlled using statistical and quantitative techniques
- ☐ Optimizing : Quantitatively managed processes are continually improved

Question A.17: Please briefly describe your duties and responsibilities:

Section : 2

PART B : Knowledge

Identify and rate the knowledge required to perform your job in the application software development project team.

Column : C1 : Rate your knowledge/skill at the time you joined the project, first time.
Please use the following rating scale.
0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert
Column : C2: Rate your knowledge/skill as of now. Use the above rating scale.
Column : C3 – To what extent this knowledge helped you in your work on the work package. Use the following rating scale
0 = nil, 1 = small, 2 = moderate, 3 = great

B.1 Technique skills

Skills that are directly required to performing the particular job. Examples : A programmer generally requires technical skills on languages, compilers, operating systems. Business analyst requires business analysis skills. Project manager needs project management skills.

Please circle the chosen option.

Knowledge/Skill	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
Project management skills	0 1 2 3	0 1 2 3	0 1 2 3
Knowledge of development methodologies used to build the application system examples : Waterfall model and spiral model	0 1 2 3	0 1 2 3	0 1 2 3
Knowledge of application radial architecture	0 1 2 3	0 1 2 3	0 1 2 3

Knowledge/Skill	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
Testing skills	0 1 2 3	0 1 2 3	0 1 2 3
Business analysis skills	0 1 2 3	0 1 2 3	0 1 2 3
Application development tool	0 1 2 3	0 1 2 3	0 1 2 3
Window design tool	0 1 2 3	0 1 2 3	0 1 2 3
App. Development tool 2	0 1 2 3	0 1 2 3	0 1 2 3
Database	0 1 2 3	0 1 2 3	0 1 2 3
Report generator	0 1 2 3	0 1 2 3	0 1 2 3
Mainframe – ISPF	0 1 2 3	0 1 2 3	0 1 2 3
Tools to access database	0 1 2 3	0 1 2 3	0 1 2 3

B.2 Knowledge of Application Software Items
Knowledge of user interface functionality, component functionality, interfaces with other systems, batch processes, database structures and overall architecture of the application system
Please circle the chosen option.

Software Item	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
Knowledge of CMS table structures of the application system	0 1 2 3	0 1 2 3	0 1 2 3
Knowledge of the overall architecture of the application	0 1 2 3	0 1 2 3	0 1 2 3
Knowledge of the software development environment at the software project	0 1 2 3	0 1 2 3	0 1 2 3

Identify and rate the knowledge required to perform your job in the application software development project team.

Column : C1 : Rate your knowledge/skill at the time you joined the project.
Please use the following rating scale.
0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert
Column : C2: Rate your knowledge/skill as of now. Use the above rating scale.
Column : C3 – To what extent this knowledge helped you in your work on the work package. Use the following rating scale
0 = nil, 1 = small, 2 = moderate, 3 = great

Your knowledge of the following application software item functionality	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
Knowledge of xxx create batch	0 1 2 3	0 1 2 3	0 1 2 3
Knowledge of xxx Case create batch	0 1 2 3	0 1 2 3	0 1 2 3
Knowledge of xxx Case create batch	0 1 2 3	0 1 2 3	0 1 2 3
Knowledge of xxx case update batch	0 1 2 3	0 1 2 3	0 1 2 3
Knowledge of xxx refresh batch	0 1 2 3	0 1 2 3	0 1 2 3
Knowledge of xxx refresh batch	0 1 2 3	0 1 2 3	0 1 2 3
…	0 1 2 3	0 1 2 3	0 1 2 3
…	0 1 2 3	0 1 2 3	0 1 2 3
…	0 1 2 3	0 1 2 3	0 1 2 3
…	0 1 2 3	0 1 2 3	0 1 2 3
…	0 1 2 3	0 1 2 3	0 1 2 3
…	0 1 2 3	0 1 2 3	0 1 2 3
…	0 1 2 3	0 1 2 3	0 1 2 3
…	0 1 2 3	0 1 2 3	0 1 2 3
…	0 1 2 3	0 1 2 3	0 1 2 3
…	0 1 2 3	0 1 2 3	0 1 2 3
Any other (Write here)	0 1 2 3	0 1 2 3	0 1 2 3

B3. Soft skills: Organisational and interpersonal (such as communication, team work) skills.

Please circle the chosen option.

Knowledge/Skill	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
Team work skills (long term)	0 1 2 3	0 1 2 3	0 1 2 3
Time management skills	0 1 2 3	0 1 2 3	0 1 2 3
Adaptability to new technologies & languages	0 1 2 3	0 1 2 3	0 1 2 3
Verbal communication skills	0 1 2 3	0 1 2 3	0 1 2 3
Inter-team communication	0 1 2 3	0 1 2 3	0 1 2 3
Investigative skills	0 1 2 3	0 1 2 3	0 1 2 3
Organisational skills	0 1 2 3	0 1 2 3	0 1 2 3
Ability to give and receive constructive criticism	0 1 2 3	0 1 2 3	0 1 2 3
Ability to multitask	0 1 2 3	0 1 2 3	0 1 2 3

Knowledge/Skill	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
Leadership skills	0 1 2 3	0 1 2 3	0 1 2 3
Negotiation skills	0 1 2 3	0 1 2 3	0 1 2 3
Interpersonal skills	0 1 2 3	0 1 2 3	0 1 2 3
Stress management skills	0 1 2 3	0 1 2 3	0 1 2 3
General writing skills	0 1 2 3	0 1 2 3	0 1 2 3
Idea initiation skills	0 1 2 3	0 1 2 3	0 1 2 3
Listening skills	0 1 2 3	0 1 2 3	0 1 2 3
Technical writing skills	0 1 2 3	0 1 2 3	0 1 2 3
Any other (Write here)	0 1 2 3	0 1 2 3	0 1 2 3

B4. Knowledge of processes and procedures: Knowledge of formal and informal processes such as promotion, review and configuration management

Please circle the chosen option.

Knowledge/Skill	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
Knowledge of software development processes used in the software project	0 1 2 3	0 1 2 3	0 1 2 3
Any other (Write here)	0 1 2 3	0 1 2 3	0 1 2 3

Please list examples of software development processes that helped you to perform your job.

B5. Business domain area knowledge: Knowledge of business domain area relating to the application system (examples: banking, insurance, government)

Please circle the chosen option.

Knowledge/Skill	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
Knowledge of xxx domain area	0 1 2 3	0 1 2 3	0 1 2 3
Any other (Write here)	0 1 2 3	0 1 2 3	0 1 2 3

Please describe the business domain area knowledge required to perform your job .

B6. Social Network knowledge: Your relationships with friends and colleagues, who can help you do the job, (within team, organisation and outside the organisation) and your understanding of 'who knows what?' among them.

Please circle the chosen option.

Knowledge/Skill	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
Understanding of 'who knows what? ' within software project team	0 1 2 3	0 1 2 3	0 1 2 3
Understanding 'who knows what?' amongst members of Environment support team	0 1 2 3	0 1 2 3	0 1 2 3
Understanding 'who knows what?' amongst members of CMS business team.	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of 'who knows what?' amongst members of CMS production support team.	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of 'who knows what' amongst various interface system teams with CMS	0 1 2 3	0 1 2 3	0 1 2 3
Professional network of people outside the organisation	0 1 2 3	0 1 2 3	0 1 2 3
Any other (Write here)	0 1 2 3	0 1 2 3	0 1 2 3

B7. General systems knowledge: Your problem-solving skills including analytical and modelling skills.

Please circle the chosen option.

Knowledge/Skill	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
General problem solving skills	0 1 2 3	0 1 2 3	0 1 2 3
Analytical and modelling skills	0 1 2 3	0 1 2 3	0 1 2 3
Any other (Write here)	0 1 2 3	0 1 2 3	0 1 2 3

Please write any comments on how general systems knowledge helped you to perform your job , if appropriate.

B8. Cultural Knowledge: Your knowledge of organisational norms, values, roles and standards of conduct that govern your interaction with your colleagues and other stakeholders

Please circle the chosen option.

Knowledge/Skill	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
Understanding of innovation and risk taking nature of organisation	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of the expectations of organisation with respect to attention to detail	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of the degree to which the management focuses on outcomes rather than the method adopted	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of the degree to which management decisions take into consideration the effect of outcomes on people	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of the degree to which work activities are organised around teams rather than individuals	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of the knowledge sharing culture within organisation	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of the aggressiveness of people in organisation	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of rites and rituals in the organisation	0 1 2 3	0 1 2 3	0 1 2 3

Knowledge/Skill	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
Understanding of leadership and vision at your organisation.	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of stories and myths in the organisation	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of power structures in the organisation	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of incentives	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of the way communication is carried out within the organisation	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of the way in which control systems work within the organisation	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of organisational structure	0 1 2 3	0 1 2 3	0 1 2 3
Understanding of hierarchy	0 1 2 3	0 1 2 3	0 1 2 3
Any other (Write here)	0 1 2 3	0 1 2 3	0 1 2 3

B9. Heuristics knowledge: Your knowledge of shortcuts, standards, guidelines, rules of thumb etc.
Please circle the chosen option.

Knowledge/Skill	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
Knowledge of shortcuts to accomplish various tasks	0 1 2 3	0 1 2 3	0 1 2 3
Knowledge of guidelines within the organisation	0 1 2 3	0 1 2 3	0 1 2 3
Knowledge of standards within the organisation	0 1 2 3	0 1 2 3	0 1 2 3
Knowledge of rules of thumb to perform various jobs	0 1 2 3	0 1 2 3	0 1 2 3
Any other (Write here)	0 1 2 3	0 1 2 3	0 1 2 3

Please write down any standards, guidelines or rules of thumb that you have used to perform the job

B10. Any other knowledge required to perform your job.
Please circle the chosen option.

Knowledge/Skill	Knowledge/Skill rating at the time you joined the project. (C1) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Current rating of knowledge/skill (C2) 0 = No knowledge, 1 – Basic, 2 – Competent, 3 - Expert	Extent to which this knowledge helped in performing the job (C3) 0 = nil, 1 = small, 2 = moderate, 3 = great
	0 1 2 3	0 1 2 3	0 1 2 3

Question B11: It is said that your actual on the job performance will improve with experience in working on a system (Examples of performance measures are efficiency, error rate, adherence to schedules and quality of your work output).
Please answer the following questions using a scale of 0 to 3 where -

0 = nil, 1 = small, 2 = moderate, 3 = great

Now, think back over your performance since joining your current software project
How much would you say your performance has improved since starting on your current project?

(Scale 0 to 3) ----------------------

If you answered ‘0-nil’ to question B11, on the previous page, you don’t need to answer this question.
Question B12: To what degree has the following factors contributed to your improved performance.
Please circle the chosen option. Please write your comments at the last column.

	not applicable	**little help**	**moderate help**	**great help**	**Comments**
Effort	•	•	•	•	
Increase in your technique skills*	•	•	•	•	
Increase in your soft skills*	•	•	•	•	
Increase in your business domain area knowledge*	•	•	•	•	
Increase in your knowledge of processes and procedures*	•	•	•	•	
Achievement needs	•	•	•	•	
Increase in your systems knowledge*	•	•	•	•	
Increase in your social network knowledge*	•	•	•	•	
Increase in your cultural knowledge*	•	•	•	•	
Increase in your heuristics knowledge*	•	•	•	•	
Increase in your knowledge of application software Items*	•	•	•	•	
Self-esteem	•	•	•	•	
Personal reasons	•	•	•	•	
Any other1(Please write)	•	•	•	•	
Any other2(Please write)	•	•	•	•	

***** - Refer next page for more details.

Your technique skills: Skills that are directly required to performing the particular job. Examples: A programmer generally requires technical skills on languages, compilers, operating systems. Business analyst requires business analysis skills. Project manager needs project management skills.

Your soft skills: Organisational and interpersonal(such as communication, team work) skills.

Business domain area knowledge: Knowledge of business domain area relating to the application system (examples: banking, insurance, government)

Your knowledge of processes and procedures: Knowledge of formal and informal processes such as promotion, review and configuration management

General systems knowledge: Your problem-solving skills including analytical and modelling skills

Social Network knowledge: Your relationships with friends and colleagues, who can help you do the job, (within team, organisation and outside the organisation) and your understanding of 'who knows what' among them.

Cultural Knowledge: Your knowledge of organisational norms, values, roles and standards of conduct that govern your interaction with your colleagues and other stakeholders

Heuristics knowledge: Your knowledge of shortcuts, standards, guidelines, rules of thumb etc.

Your knowledge of Application Software Items: Knowledge of user interface functionality, component functionality, interfaces with other systems, batch processes, database structures and overall architecture of the application system

11.4 General-Survey Questionnaire

Knowledge and Performance: Interview questionnaire

This survey may take approximately 20 to 30 minutes to complete. Thank you for taking the time to complete this survey!

PART A: DEMOGRAPHICS AND WORK EXPERIENCE

Question A.1: How long have you been working in application software delivery projects?(in years and months) __________

Question A.2: What is your gender? ☐ **Male** ☐ **Female**

Question A.3: What is your mode of work?

☐ Predominantly work at the organisation's location with other members of the team.
☐ Work remotely from home
☐ Any other , please specify. ___________________________________

Question A.4: What is the highest qualification you have obtained?

☐ High school certificate ☐ Vocational Certificate ☐ Diploma ☐ Bachelor's degree ☐ Master's Degree/Post Graduate ☐ Doctorate
Other (specify) _______________

Question A.5: What is the field of study for your highest qualification? (e.g. Maths, Computer science) _________________________

Question A.6: What is your main role?

☐ Business Analyst ☐ Programmer ☐ Programmer/Analyst ☐ Architect ☐ Designer ☐ Tester ☐ Project Manager ☐ Other (Please specify)______________________

The following questions relate to the application software project team and the application software system with which you are currently working.

Question A.7: How long have you been working with this software project team? (in years & months) ________

Question A.8: How many years have you been working on this application software system? (in years & months) ________

Question A.9: Please write the name of the city in which you worked, for the longest time, during this software project. __________________

Question A.10: What is your main role in this software project team?

☐ Business Analyst ☐ Programmer ☐ Programmer/Analyst ☐ Architect ☐ Designer ☐ Tester ☐ Project Manager
☐ Other (Please specify)______________________

Question A.11: What was the industry segment of the application software? (eg. Insurance, transport, Banking, Government, etc) ___________________________

Question A.12: Indicate the relative time you have spent working on the various aspects of your current work. Circle the appropriate option.

	not applicable	very rarely	rarely	regularly	very regularly	most of the time
Requirements analysis	•	•	•	•	•	•
Architectural design	•	•	•	•	•	•
High level design	•	•	•	•	•	•
Low level design	•	•	•	•	•	•
Coding and unit testing	•	•	•	•	•	•
Function testing	•	•	•	•	•	•
Integration testing	•	•	•	•	•	•
Configuration management	•	•	•	•	•	•
Quality assurance	•	•	•	•	•	•
Preparation of manuals	•	•	•	•	•	•
Implementation	•	•	•	•	•	•
Production support	•	•	•	•	•	•
Project Management	•	•	•	•	•	•
Other-1 (write here)	•	•	•	•	•	•
Other-2 (write here)	•	•	•	•	•	•

Question A.13: Does your organisation adhere to any type of software process model or standard?

- [ ] Capability Maturity Model Integrated (CMMI)
- [ ] Software Process Improvement and Capability dEtermination (SPICE)
- [ ] International Standards Organisation (ISO-9001)
- [ ] Rational Unified process (RUP)
- [ ] PRINCE2
- [ ] Any other , please specify. ____

Question A.14: If your organisation adheres to any quality process model/standard, write down any the formal accreditation/certification levels on quality.

Question A.15: What is your opinion on software development processes in your software project team?

- [ ] No documentation exists for software development processes. [ ] Software development processes are documented
- [ ] Documented processes are followed and managed occasionally.
- [ ] Processes are well understood and implemented by everyone, and are described in standards, procedures, tools, and methods
- [ ] Metrics are collected on the managed processes and controlled using statistical and quantitative techniques
- [ ] Quantitatively managed processes are continually improved

Question A.16: Does your project team has defined standards and guidelines?

[ ] Yes [ ] No

Question A.17: Please briefly describe your duties and responsibilities:

Question A.18: Please briefly list the technologies used to develop the application software system in your software project.

Question A.19: Does your organisation have any tools to locate people with specific knowledge?

PART B: Knowledge items required to perform your job.

Please indicate the extent to which the following items help you to perform your main role successfully

Please circle the chosen option for each item.

	not applicable	**little help**	**moderate help**	**great help**	**necessary**
Team work skills (long term)	•	•	•	•	•
Time management skills	•	•	•	•	•
Adaptability to new technologies & languages	•	•	•	•	•
Verbal communication skills	•	•	•	•	•
Inter-team communication	•	•	•	•	•
Investigative skills	•	•	•	•	•
Organisational skills	•	•	•	•	•
Ability to give and receive constructive criticism	•	•	•	•	•
Ability to multitask	•	•	•	•	•
Leadership skills	•	•	•	•	•
Negotiation skills	•	•	•	•	•

Please indicate the extent to which the following items help you to perform your main role successfully

	not applicable	little help	moderate help	great help	necessary
Interpersonal skills	•	•	•	•	•
Stress management skills	•	•	•	•	•
General writing skills	•	•	•	•	•
Idea initiation skills	•	•	•	•	•
Listening skills	•	•	•	•	•
Technical writing skills	•	•	•	•	•
Software skills required to develop the application software system The specialised knowledge and skills required to work in a software development team such as the knowledge of computer languages, operating systems, data bases	•	•	•	•	•
Hardware skills required to develop the application software system	•	•	•	•	•
Business analysis skills	•	•	•	•	•
Testing skills	•	•	•	•	•
Project management skills	•	•	•	•	•
Business domain area knowledge relating to the application system examples: banking, insurance and shipping	•	•	•	•	•
Knowledge of software development processes used in the software project	•	•	•	•	•

Please indicate the extent to which the following items help you to perform your main role successfully

	not applicable	little help	moderate help	great help	necessary
Knowledge of application software architectures examples: client-server, J2EE, .NET, radial architecture, information architecture	•	•	•	•	•
Understanding of ‘who knows what? ’ within software project team	•	•	•	•	•
As above, within organisation but outside software project team	•	•	•	•	•
General problem solving skills	•	•	•	•	•
Analytical and modelling skills	•	•	•	•	•
Knowledge of development methodologies used to build the application system examples: Waterfall model and spiral model	•	•	•	•	•
Professional network of people outside the organisation	•	•	•	•	•
Understanding of innovation and risk taking nature of organisation	•	•	•	•	•
Understanding of the expectations of organisation with respect to attention to detail	•	•	•	•	•
Understanding of the degree to which the management focuses on outcomes rather than the method adopted	•	•	•	•	•

Please indicate the extent to which the following items help you to perform your main role successfully

	not applicable	little help	moderate help	great help	necessary
Understanding of the degree to which management decisions take into consideration the effect of outcomes on people	•	•	•	•	•
Understanding of the degree to which work activities are organised around teams rather than individuals	•	•	•	•	•
Understanding of the knowledge sharing culture within organisation	•	•	•	•	•
Understanding of the aggressiveness of people in organisation	•	•	•	•	•
Understanding of the degree to which organisational activities emphasise maintaining the status quo in contrast to growth	•	•	•	•	•
Knowledge of shortcuts to accomplish various tasks	•	•	•	•	•
Knowledge of guidelines within the organisation	•	•	•	•	•
Knowledge of standards within the organisation	•	•	•	•	•
Knowledge of rules of thumb to perform various jobs	•	•	•	•	•
Knowledge of user interfaces within the application system	•	•	•	•	•
Knowledge of the interfaces of the application system with other systems	•	•	•	•	•
Knowledge of program functionality within the system	•	•	•	•	•
Knowledge of any reusable component functionality within the application system	•	•	•	•	•

Please indicate the extent to which the following items help you to perform your main role successfully

	not applicable	little help	moderate help	great help	necessary
Knowledge of database structures of the application system	•	•	•	•	•
Knowledge of the overall architecture of the application	•	•	•	•	•
Knowledge of the software development environment at the software project	•	•	•	•	•

Please briefly describe any other type of knowledge that helped to perform your job and its relative importance level.
____
____
____

PART C: Why performance increases over time.

It is said that your actual on the job performance will improve with experience in working on a system (examples of performance measures are efficiency, error rate, adherence to schedules, quality of your work output).

Please answer the following questions using a scale of 0 to 3 where -

0 = nil, 1 = small, 2 = moderate, 3 = great

Now, think back over your performance since joining your current software project

Question C.1: How much would you say your job performance has improved since starting on your current project? (scale 0 to 3) ---------------------

Question C.2: Which types of knowledge and skill have contributed to this improvement?

Column A: Indicate the increase in each type of your skill/knowledge since joining the software project team.
Column B: Indicate the relative contribution each type of knowledge made to your increase in performance.

No.	Operational Knowledge Type	**Column A** Relative increase in this type of knowledge (Use a scale of 0 to 3)	**Column B** Relative contribution to increase in your performance (Use a scale of 0 to 3)
1.	**Your technique skills:** Skills that are directly required to performing the particular job. Examples: A programmer generally requires technical skills on languages, compilers, operating systems. Business analyst requires business analysis skills. Project manager needs project management skills.		
2.	**Your soft skills:** Organisational and interpersonal(such as communication, team work) skills.		
3.	**Business domain area knowledge:** Knowledge of business domain area relating to the application system (examples: banking, insurance, government)		
4.	**Your knowledge of processes and procedures:** Knowledge of formal and informal processes such as promotion, review and configuration management		
5.	**General systems knowledge:** Your problem-solving skills including analytical and modelling skills		
6.	**Social Network knowledge:** Your relationships with friends and colleagues, who can help you do the job, (within team, organisation and outside the organisation) and your understanding of 'who knows what' among them.		
7.	**Cultural Knowledge:** Your knowledge of organisational norms, values, roles and standards of conduct that govern your interaction with your colleagues and other stakeholders		
8.	**Heuristics knowledge:** Your knowledge of shortcuts, standards, guidelines, rules of thumb etc.		
9.	**Your knowledge of Application Software Items:** Knowledge of user interface functionality, component functionality, interfaces with other systems, batch processes, database structures and overall architecture of the application system		
10.	**Any other type of knowledge (briefly describe here)**		

Question C3: If possible, give one significant example of how any of the types of knowledge contributed to your increased performance.

(Survey continued on next page)

Question C4: The results of this study can be notified using email. Do you wish to receive the results of this survey? If so, please write down your email address.

11.5 Relation between Total Professional Experience and contribution to performance

11.5.1.1 Relation : Total Professional Experience and perceived contribution of increase in process knowledge to increase in performance

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.193[a]	.037	.031	.8600

a. Predictors: (Constant), Total Professional Experience

b. Dependent Variable: C24B.Cntrbtn. of process kno. to inc. in performance

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	4.358	1	4.358	5.892	.016[a]
	Residual	112.407	152	.740		
	Total	116.765	153			

a. Predictors: (Constant), Total Professional Experience

b. Dependent Variable: C24B.Cntrbtn. of process kno. to inc. in performance

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.010	.117		17.198	.000
	Total Professional Experience	-.023	.009	-.193	-2.427	.016

a. Dependent Variable: C24B.Cntrbtn. of process kno. to inc. in performance

11.5.1.2 Relation : Total Professional Experience and contribution of systems knowledge to increase in performance

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.172[a]	.029	.023	.8439

a. Predictors: (Constant), Total Professional Experience

b. Dependent Variable: C25B.Cntrbtn. of systems kno. to inc. in performance

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.333	1	3.333	4.679	.032[a]
	Residual	109.676	154	.712		
	Total	113.008	155			

a. Predictors: (Constant), Total Professional Experience

b. Dependent Variable: C25B.Cntrbtn. of systems kno. to inc. in performance

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.260	.114		19.837	.000
	Total Professional Experience	-.020	.009	-.172	-2.163	.032

a. Dependent Variable: C25B.Cntrbtn. of systems kno. to inc. in performance

11.5.1.3 Relation : Total Professional Experience and contribution of cultural knowledge to increase in performance

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.183[a]	.034	.027	.8209

a. Predictors: (Constant), Total Professional Experience

b. Dependent Variable: C27B.Cntrbtn. of cultural kno. to inc. in performance

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.583	1	3.583	5.317	.022[a]
	Residual	103.105	153	.674		
	Total	106.688	154			

a. Predictors: (Constant), Total Professional Experience

b. Dependent Variable: C27B.Cntrbtn. of cultural kno. to inc. in performance

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	1.890	.111		16.995	.000
	Total Professional Experience	-.021	.009	-.183	-2.306	.022

a. Dependent Variable: C27B.Cntrbtn. of cultural kno. to inc. in performance

11.6 Relation : Experience within SE team and contribution of knowledge types to performance

11.6.1.1 Relation between Experience within SE team and perceived contribution of increase in technique skills to improved performance

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.264[a]	.069	.064	.8232

a. Predictors: (Constant), Experience in Software Engineering Team

b. Dependent Variable: C21B.Cntrbtn. of Tech.skills to inc. in performance

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	8.041	1	8.041	11.864	.001[a]
	Residual	107.760	159	.678		
	Total	115.801	160			

a. Predictors: (Constant), Experience in Software Engineering Team

b. Dependent Variable: C21B.Cntrbtn. of Tech.skills to inc. in performance

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	1.943	.091		21.415	.000
	Experience in Software Engineering Team	.084	.024	.264	3.444	.001

a. Dependent Variable: C21B.Cntrbtn. of Tech.skills to inc. in performance

11.6.1.2 Relation: Experience within SE team and contribution of increase in configuration knowledge to increase in performance.

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.182[a]	.033	.027	.8158

a. Predictors: (Constant), Experience in Software Engineering Team

b. Dependent Variable: C29B.Cntrbtn. of config. kno. to inc. in performance

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.601	1	3.601	5.412	.021[a]
	Residual	105.142	158	.665		
	Total	108.744	159			

a. Predictors: (Constant), Experience in Software Engineering Team

b. Dependent Variable: C29B.Cntrbtn. of config. kno. to inc. in performance

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	1.972	.090		21.867	.000
	Experience in Software Engineering Team	.057	.024	.182	2.326	.021

a. Dependent Variable: C29B.Cntrbtn. of config. kno. to inc. in performance

11.7 ANOVA results: Demographic profile information and re-classified factors

11.7.1.1 Defined standards and guidelines and Management skills

Tests of Between-Subjects Effects

Dependent Variable: x3.Management Skills

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Corrected Model	3.976[a]	3	1.325	4.578	.004
Intercept	144.028	1	144.028	497.523	.000
A.16DefinedStandards	3.976	3	1.325	4.578	.004
Error	50.371	174	.289		
Total	1531.797	178			
Corrected Total	54.347	177			

a. R Squared = .073 (Adjusted R Squared = .057)

Estimates

Dependent Variable: x3.Management Skills

A16.Defined Standards	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
	3.042	.269	2.511	3.573
No	2.631	.097	2.441	2.822
Somewhat	2.045	.380	1.295	2.796
Yes	2.943	.045	2.854	3.033

11.7.1.2 Defined standards and guidelines and heuristics

Tests of Between-Subjects Effects

Dependent Variable: x4.Heuristics

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Corrected Model	6.040[a]	3	2.013	4.341	.006
Intercept	65.327	1	65.327	140.854	.000
A.16DefinedStandards	6.040	3	2.013	4.341	.006
Error	79.772	172	.464		
Total	1390.361	176			
Corrected Total	85.812	175			

a. R Squared = .070 (Adjusted R Squared = .054)

Estimates

Dependent Variable: x4.Heuristics

A16.Defined Standards	Mean	Std. Error	95% Confidence Interval Lower Bound	95% Confidence Interval Upper Bound
	2.938	.341	2.265	3.610
No	2.452	.122	2.210	2.693
Somewhat	1.000	.681	-.344	2.344
Yes	2.789	.058	2.675	2.902

11.7.1.3 Defined standards and guidelines and business functional KSA

Tests of Between-Subjects Effects

Dependent Variable: x6.Business Functional KSA

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Corrected Model	3.095[a]	3	1.032	2.894	.037
Intercept	153.574	1	153.574	430.770	.000
A.16DefinedStandards	3.095	3	1.032	2.894	.037
Error	62.033	174	.357		
Total	1612.797	178			
Corrected Total	65.128	177			

a. R Squared = .048 (Adjusted R Squared = .031)

Estimates

Dependent Variable: x6.Business Functional KSA

			95% Confidence Interval	
A16.Defined Standards	Mean	Std. Error	Lower Bound	Upper Bound
	3.229	.299	2.640	3.818
No	2.792	.107	2.581	3.004
Somewhat	2.000	.422	1.167	2.833
Yes	2.989	.050	2.889	3.088

11.7.1.4 Field of study and communication and relationship skills

Tests of Between-Subjects Effects

Dependent Variable: x5.Communication and relationship skills

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Corrected Model	4.535[a]	6	.756	2.592	.020
Intercept	319.576	1	319.576	1095.697	.000
A5RecodedFieldOfStudy	4.535	6	.756	2.592	.020
Error	46.375	159	.292		
Total	1804.935	166			
Corrected Total	50.910	165			

a. R Squared = .089 (Adjusted R Squared = .055)

Estimates

Dependent Variable: x5.Communication and relationship skills

A5.FieldOfStudy	Mean	Std. Error	95% Confidence Interval Lower Bound	95% Confidence Interval Upper Bound
1.00 InformationTechnology Related topics	3.255	.051	3.155	3.355
2.00 Mathematics	3.470	.242	2.993	3.947
4.00 Media	3.200	.540	2.133	4.267
5.00 Business administration	3.571	.144	3.286	3.856
6.00 Engineering	3.307	.139	3.031	3.582
7.00 Science	2.889	.180	2.533	3.244
8.00 Others	2.800	.191	2.423	3.177

11.7.1.5 Field of study and communication and Software development KSA

Tests of Between-Subjects Effects

Dependent Variable: x2.Software Development KSA

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Corrected Model	10.018[a]	6	1.670	5.179	.000
Intercept	230.395	1	230.395	714.577	.000
A5RecodedFieldOfStudy	10.018	6	1.670	5.179	.000
Error	51.265	159	.322		
Total	1594.046	166			
Corrected Total	61.283	165			

a. R Squared = .163 (Adjusted R Squared = .132)

Estimates

Dependent Variable: x2.Software Development KSA

A5.FieldOfStudy	Mean	Std. Error	95% Confidence Interval Lower Bound	95% Confidence Interval Upper Bound
1.00 InformationTechnology Related topics	3.136	.053	3.031	3.241
2.00 Mathematics	3.125	.254	2.623	3.627
4.00 Media	1.667	.568	.545	2.788
5.00 Business administration	2.458	.152	2.159	2.758
6.00 Engineering	3.252	.147	2.963	3.542
7.00 Science	2.756	.189	2.382	3.130
8.00 Others	2.703	.201	2.307	3.100

11.7.1.6 Gender and Management skills

Tests of Between-Subjects Effects

Dependent Variable: x3.Management Skills

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Corrected Model	1.459[a]	1	1.459	4.854	.029
Intercept	1244.222	1	1244.222	4140.440	.000
A.2Gender	1.459	1	1.459	4.854	.029
Error	52.889	176	.301		
Total	1531.797	178			
Corrected Total	54.347	177			

a. R Squared = .027 (Adjusted R Squared = .021)

Estimates

Dependent Variable: x3.Management Skills

A2.Gender	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
female	3.024	.077	2.872	3.175
male	2.824	.049	2.728	2.920

11.7.1.7 Gender and Software development KSA

Tests of Between-Subjects Effects

Dependent Variable: x2.Software Development KSA

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Corrected Model	2.494[a]	1	2.494	6.778	.010
Intercept	1292.436	1	1292.436	3512.629	.000
A.2Gender	2.494	1	2.494	6.778	.010
Error	64.757	176	.368		
Total	1707.672	178			
Corrected Total	67.251	177			

a. R Squared = .037 (Adjusted R Squared = .032)

Estimates

Dependent Variable: x2.Software Development KSA

A2.Gender	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
female	2.849	.085	2.681	3.017
male	3.111	.054	3.005	3.217

11.7.1.8 Knowledge locator tools and Organisational culture and relationships

Tests of Between-Subjects Effects

Dependent Variable: x1.Org Culture

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Corrected Model	3.676[a]	3	1.225	2.742	.045
Intercept	814.193	1	814.193	1821.894	.000
A. 19KnowledgeLocator Tools	3.676	3	1.225	2.742	.045
Error	77.313	173	.447		
Total	1122.836	177			
Corrected Total	80.989	176			

a. R Squared = .045 (Adjusted R Squared = .029)

Estimates

Dependent Variable: x1.Org Culture

A19.Knowledge Locator Tools	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
	2.471	.115	2.245	2.697
No	2.288	.101	2.089	2.487
Unsure	2.172	.143	1.890	2.453
Yes	2.558	.076	2.408	2.708

11.8 SDLC Phases and knowledge factors

11.8.1.1 Requirements analysis

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12a.Requirements Analysis[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x1.Org Culture

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.239[a]	.057	.052	.66062	2.151

a. Predictors: (Constant), A12a.Requirements Analysis

b. Dependent Variable: x1.Org Culture

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	4.354	1	4.354	9.976	.002[a]
	Residual	71.573	164	.436		
	Total	75.927	165			

a. Predictors: (Constant), A12a.Requirements Analysis

b. Dependent Variable: x1.Org Culture

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.112	.112		18.885	.000
	A12a.Requirements Analysis	.110	.035	.239	3.159	.002

a. Dependent Variable: x1.Org Culture

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.1122	2.6618	2.4261	.16244	167
Residual	-1.56177	1.44815	-.01747	.64459	166
Std. Predicted Value	-1.933	1.451	.000	1.000	167
Std. Residual	-2.364	2.192	-.026	.976	166

a. Dependent Variable: x1.Org Culture

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12a.Requirements Analysis[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x3.Management Skills

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.255[a]	.065	.059	.53748	1.885

a. Predictors: (Constant), A12a.Requirements Analysis

b. Dependent Variable: x3.Management Skills

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.304	1	3.304	11.437	.001[a]
	Residual	47.666	165	.289		
	Total	50.970	166			

a. Predictors: (Constant), A12a.Requirements Analysis

b. Dependent Variable: x3.Management Skills

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.608	.091		28.751	.000
	A12a.Requirements Analysis	.095	.028	.255	3.382	.001

a. Dependent Variable: x3.Management Skills

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.6083	3.0857	2.8810	.14108	167
Residual	-1.44475	1.20072	-.01087	.53780	167
Std. Predicted Value	-1.933	1.451	.000	1.000	167
Std. Residual	-2.688	2.234	-.020	1.001	167

a. Dependent Variable: x3.Management Skills

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12a.Requirements Analysis[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x4.Heuristics

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.178[a]	.032	.026	.69121	1.916

a. Predictors: (Constant), A12a.Requirements Analysis

b. Dependent Variable: x4.Heuristics

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	2.540	1	2.540	5.317	.022[a]
	Residual	77.878	163	.478		
	Total	80.418	164			

a. Predictors: (Constant), A12a.Requirements Analysis

b. Dependent Variable: x4.Heuristics

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.482	.117		21.145	.000
	A12a.Requirements Analysis	.084	.037	.178	2.306	.022

a. Dependent Variable: x4.Heuristics

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.4820	2.9031	2.7225	.12446	167
Residual	-1.81886	1.51800	-.01784	.68477	165
Std. Predicted Value	-1.933	1.451	.000	1.000	167
Std. Residual	-2.631	2.196	-.026	.991	165

a. Dependent Variable: x4.Heuristics

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12a.Requirements Analysis[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x5.Communication and relationship skills

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.298[a]	.089	.083	.53450	1.788

a. Predictors: (Constant), A12a.Requirements Analysis

b. Dependent Variable: x5.Communication and relationship skills

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	4.588	1	4.588	16.059	.000[a]
	Residual	47.140	165	.286		
	Total	51.728	166			

a. Predictors: (Constant), A12a.Requirements Analysis

b. Dependent Variable: x5.Communication and relationship skills

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.934	.090		32.518	.000
	A12a.Requirements Analysis	.112	.028	.298	4.007	.000

a. Dependent Variable: x5.Communication and relationship skills

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.9337	3.4962	3.2551	.16625	167
Residual	-1.47122	1.06626	-.01583	.53494	167
Std. Predicted Value	-1.933	1.451	.000	1.000	167
Std. Residual	-2.752	1.995	-.030	1.001	167

a. Dependent Variable: x5.Communication and relationship skills

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12a.Requirements Analysis[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x6.Business Functional KSA

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.290[a]	.084	.079	.58220	1.995

a. Predictors: (Constant), A12a.Requirements Analysis

b. Dependent Variable: x6.Business Functional KSA

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	5.152	1	5.152	15.200	.000[a]
	Residual	55.928	165	.339		
	Total	61.081	166			

a. Predictors: (Constant), A12a.Requirements Analysis

b. Dependent Variable: x6.Business Functional KSA

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.608	.098		26.541	.000
	A12a.Requirements Analysis	.119	.031	.290	3.899	.000

a. Dependent Variable: x6.Business Functional KSA

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.6082	3.2042	2.9487	.17617	167
Residual	-1.80424	1.22514	.00920	.57109	167
Std. Predicted Value	-1.933	1.451	.000	1.000	167
Std. Residual	-3.099	2.104	.016	.981	167

a. Dependent Variable: x6.Business Functional KSA

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12a.Requirements Analysis[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x7.Problem solving abilities

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.212[a]	.045	.039	.54973	1.947

a. Predictors: (Constant), A12a.Requirements Analysis

b. Dependent Variable: x7.Problem solving abilities

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	2.348	1	2.348	7.769	.006[a]
	Residual	49.863	165	.302		
	Total	52.211	166			

a. Predictors: (Constant), A12a.Requirements Analysis

b. Dependent Variable: x7.Problem solving abilities

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.987	.093		32.196	.000
	A12a.Requirements Analysis	.080	.029	.212	2.787	.006

a. Dependent Variable: x7.Problem solving abilities

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.9875	3.3898	3.2173	.11892	167
Residual	-1.64842	1.01251	-.01274	.54467	167
Std. Predicted Value	-1.933	1.451	.000	1.000	167
Std. Residual	-2.999	1.842	-.023	.991	167

a. Dependent Variable: x7.Problem solving abilities

11.8.1.2 Architectural design

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12b.Architectural design[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x1.Org Culture

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.178[a]	.032	.025	.66965	1.952

a. Predictors: (Constant), A12b.Architectural design

b. Dependent Variable: x1.Org Culture

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	2.242	1	2.242	5.001	.027[a]
	Residual	68.162	152	.448		
	Total	70.405	153			

a. Predictors: (Constant), A12b.Architectural design

b. Dependent Variable: x1.Org Culture

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.270	.088		25.776	.000
	A12b.Architectural design	.081	.036	.178	2.236	.027

a. Dependent Variable: x1.Org Culture

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.2705	2.6767	2.4261	.12106	155
Residual	-1.43295	1.64830	.00313	.64986	154
Std. Predicted Value	-1.286	2.070	.000	1.000	155
Std. Residual	-2.140	2.461	.005	.970	154

a. Dependent Variable: x1.Org Culture

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12b.Architectural design[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x2.Software Development KSA

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.239[a]	.057	.051	.60047	1.911

a. Predictors: (Constant), A12b.Architectural design

b. Dependent Variable: x2.Software Development KSA

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.346	1	3.346	9.280	.003[a]
	Residual	55.167	153	.361		
	Total	58.512	154			

a. Predictors: (Constant), A12b.Architectural design

b. Dependent Variable: x2.Software Development KSA

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.846	.079		36.152	.000
	A12b.Architectural design	.099	.032	.239	3.046	.003

a. Dependent Variable: x2.Software Development KSA

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.8462	3.3408	3.0358	.14740	155
Residual	-2.14298	1.15379	.00450	.59332	155
Std. Predicted Value	-1.286	2.070	.000	1.000	155
Std. Residual	-3.569	1.921	.007	.988	155

a. Dependent Variable: x2.Software Development KSA

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12b.Architectural design[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x5.Communication and relationship skills

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.261[a]	.068	.062	.54057	1.577

a. Predictors: (Constant), A12b.Architectural design

b. Dependent Variable: x5.Communication and relationship skills

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.280	1	3.280	11.224	.001[a]
	Residual	44.709	153	.292		
	Total	47.988	154			

a. Predictors: (Constant), A12b.Architectural design

b. Dependent Variable: x5.Communication and relationship skills

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	3.067	.071		43.279	.000
	A12b.Architectural design	.098	.029	.261	3.350	.001

a. Dependent Variable: x5.Communication and relationship skills

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	3.0674	3.5571	3.2551	.14594	155
Residual	-1.46327	.93261	-.03731	.54676	155
Std. Predicted Value	-1.286	2.070	.000	1.000	155
Std. Residual	-2.707	1.725	-.069	1.011	155

a. Dependent Variable: x5.Communication and relationship skills

11.8.1.3 High level design

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12c.High level design[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x1.Org Culture

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.156[a]	.024	.018	.67214	1.957

a. Predictors: (Constant), A12c.High level design

b. Dependent Variable: x1.Org Culture

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.785	1	1.785	3.952	.049[a]
	Residual	71.381	158	.452		
	Total	73.166	159			

a. Predictors: (Constant), A12c.High level design

b. Dependent Variable: x1.Org Culture

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.239	.108		20.706	.000
	A12c.High level design	.073	.037	.156	1.988	.049

a. Dependent Variable: x1.Org Culture

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.2389	2.6047	2.4261	.10596	161
Residual	-1.50471	1.61476	-.00372	.65826	160
Std. Predicted Value	-1.767	1.685	.000	1.000	161
Std. Residual	-2.239	2.402	-.006	.979	160

a. Dependent Variable: x1.Org Culture

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12c.High level design[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x5.
Communication and relationship skills

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.240[a]	.058	.052	.54356	1.897

a. Predictors: (Constant), A12c.High level design

b. Dependent Variable: x5.Communication and relationship skills

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	2.881	1	2.881	9.750	.002[a]
	Residual	46.977	159	.295		
	Total	49.858	160			

a. Predictors: (Constant), A12c.High level design

b. Dependent Variable: x5.Communication and relationship skills

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	3.018	.087		34.622	.000
	A12c.High level design	.093	.030	.240	3.122	.002

a. Dependent Variable: x5.Communication and relationship skills

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	3.0180	3.4812	3.2551	.13418	161
Residual	-1.40327	.98199	-.02307	.54142	161
Std. Predicted Value	-1.767	1.685	.000	1.000	161
Std. Residual	-2.582	1.807	-.042	.996	161

a. Dependent Variable: x5.Communication and relationship skills

11.8.1.4 Low level design

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12d.Low level design[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x2.Software Development KSA

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.295[a]	.087	.081	.59084	1.939

a. Predictors: (Constant), A12d.Low level design

b. Dependent Variable: x2.Software Development KSA

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	5.163	1	5.163	14.789	.000[a]
	Residual	54.110	155	.349		
	Total	59.272	156			

a. Predictors: (Constant), A12d.Low level design

b. Dependent Variable: x2.Software Development KSA

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.680	.104		25.779	.000
	A12d.Low level design	.132	.034	.295	3.846	.000

a. Dependent Variable: x2.Software Development KSA

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.6795	3.3406	3.0358	.18192	157
Residual	-1.81175	1.32047	.01783	.58666	157
Std. Predicted Value	-1.958	1.676	.000	1.000	157
Std. Residual	-3.066	2.235	.030	.993	157

a. Dependent Variable: x2.Software Development KSA

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12d.Low level design[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x6.Business Functional KSA

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.201[a]	.041	.034	.59607	1.960

a. Predictors: (Constant), A12d.Low level design

b. Dependent Variable: x6.Business Functional KSA

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	2.329	1	2.329	6.554	.011[a]
	Residual	55.072	155	.355		
	Total	57.401	156			

a. Predictors: (Constant), A12d.Low level design

b. Dependent Variable: x6.Business Functional KSA

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.709	.105		25.838	.000
	A12d.Low level design	.089	.035	.201	2.560	.011

a. Dependent Variable: x6.Business Functional KSA

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.7094	3.1534	2.9487	.12218	157
Residual	-1.48704	1.20176	.00333	.58296	157
Std. Predicted Value	-1.958	1.676	.000	1.000	157
Std. Residual	-2.495	2.016	.006	.978	157

a. Dependent Variable: x6.Business Functional KSA

11.8.1.5 Coding and unit testing

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12e.Coding and unit testing[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x2.Software Development KSA

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.402[a]	.161	.156	.56623	2.025

a. Predictors: (Constant), A12e.Coding and unit testing

b. Dependent Variable: x2.Software Development KSA

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	9.755	1	9.755	30.426	.000[a]
	Residual	50.657	158	.321		
	Total	60.412	159			

a. Predictors: (Constant), A12e.Coding and unit testing

b. Dependent Variable: x2.Software Development KSA

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.600	.091		28.655	.000
	A12e.Coding and unit testing	.143	.026	.402	5.516	.000

a. Dependent Variable: x2.Software Development KSA

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.6003	3.3156	3.0358	.24769	160
Residual	-1.74340	1.27465	.02541	.56715	160
Std. Predicted Value	-1.758	1.130	.000	1.000	160
Std. Residual	-3.079	2.251	.045	1.002	160

a. Dependent Variable: x2.Software Development KSA

11.8.1.6 Function testing

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12f.Function testing[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x6.Business Functional KSA

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.199[a]	.040	.034	.59634	1.848

a. Predictors: (Constant), A12f.Function testing

b. Dependent Variable: x6.Business Functional KSA

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	2.328	1	2.328	6.547	.011[a]
	Residual	56.544	159	.356		
	Total	58.873	160			

a. Predictors: (Constant), A12f.Function testing

b. Dependent Variable: x6.Business Functional KSA

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.
1	(Constant)	2.742	.093		29.384	.000
	A12f.Function testing	.078	.031	.199	2.559	.011

a. Dependent Variable: x6.Business Functional KSA

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.7424	3.1341	2.9487	.12063	161
Residual	-1.73408	1.17928	.02564	.58810	161
Std. Predicted Value	-1.710	1.537	.000	1.000	161
Std. Residual	-2.908	1.978	.043	.986	161

a. Dependent Variable: x6.Business Functional KSA

11.8.1.7 Integration testing

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12g.Integration testing[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x6.Business Functional KSA

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.162[a]	.026	.020	.60047	1.885

a. Predictors: (Constant), A12g.Integration testing

b. Dependent Variable: x6.Business Functional KSA

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.537	1	1.537	4.261	.041[a]
	Residual	56.968	158	.361		
	Total	58.505	159			

a. Predictors: (Constant), A12g.Integration testing

b. Dependent Variable: x6.Business Functional KSA

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients		
		B	Std. Error	Beta	t	Sig.
1	(Constant)	2.811	.082		34.313	.000
	A12g.Integration testing	.062	.030	.162	2.064	.041

a. Dependent Variable: x6.Business Functional KSA

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.8109	3.1188	2.9487	.09830	160
Residual	-1.71885	1.12754	.01089	.58396	160
Std. Predicted Value	-1.402	1.731	.000	1.000	160
Std. Residual	-2.863	1.878	.018	.973	160

a. Dependent Variable: x6.Business Functional KSA

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12g.Integration testing[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x2.Software Development KSA

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.223[a]	.050	.044	.60278	2.068

a. Predictors: (Constant), A12g.Integration testing

b. Dependent Variable: x2.Software Development KSA

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.003	1	3.003	8.265	.005[a]
	Residual	57.409	158	.363		
	Total	60.412	159			

a. Predictors: (Constant), A12g.Integration testing

b. Dependent Variable: x2.Software Development KSA

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.843	.082		34.572	.000
	A12g.Integration testing	.086	.030	.223	2.875	.005

a. Dependent Variable: x2.Software Development KSA

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.8431	3.2736	3.0358	.13743	160
Residual	-2.01531	1.07080	.00666	.59739	160
Std. Predicted Value	-1.402	1.731	.000	1.000	160
Std. Residual	-3.343	1.776	.011	.991	160

a. Dependent Variable: x2.Software Development KSA

11.8.1.8 Configuration management

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12h.Configuration management[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x2.Software Development KSA

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.284[a]	.081	.075	.59293	2.020

a. Predictors: (Constant), A12h.Configuration management

b. Dependent Variable: x2.Software Development KSA

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	4.752	1	4.752	13.517	.000[a]
	Residual	54.140	154	.352		
	Total	58.892	155			

a. Predictors: (Constant), A12h.Configuration management

b. Dependent Variable: x2.Software Development KSA

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.838	.072		39.555	.000
	A12h.Configuration management	.113	.031	.284	3.677	.000

a. Dependent Variable: x2.Software Development KSA

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.8380	3.4010	3.0358	.17510	156
Residual	-1.83797	1.16203	.01897	.59233	156
Std. Predicted Value	-1.130	2.086	.000	1.000	156
Std. Residual	-3.100	1.960	.032	.999	156

a. Dependent Variable: x2.Software Development KSA

11.8.1.9 Quality assurance

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12i.Quality assurance[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x1.Org Culture

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.158[a]	.025	.019	.67199	2.011

a. Predictors: (Constant), A12i.Quality assurance

b. Dependent Variable: x1.Org Culture

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.809	1	1.809	4.007	.047[a]
	Residual	70.896	157	.452		
	Total	72.706	158			

a. Predictors: (Constant), A12i.Quality assurance

b. Dependent Variable: x1.Org Culture

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.253	.101		22.208	.000
	A12i.Quality assurance	.072	.036	.158	2.002	.047

a. Dependent Variable: x1.Org Culture

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.2533	2.6115	2.4261	.10701	160
Residual	-1.46823	1.74669	.01400	.65679	159
Std. Predicted Value	-1.615	1.732	.000	1.000	160
Std. Residual	-2.185	2.599	.021	.977	159

a. Dependent Variable: x1.Org Culture

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12i.Quality assurance[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x6.Business Functional KSA

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.179[a]	.032	.026	.59873	1.792

a. Predictors: (Constant), A12i.Quality assurance

b. Dependent Variable: x6.Business Functional KSA

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.866	1	1.866	5.204	.024[a]
	Residual	56.639	158	.358		
	Total	58.505	159			

a. Predictors: (Constant), A12i.Quality assurance

b. Dependent Variable: x6.Business Functional KSA

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.774	.090		30.779	.000
	A12i.Quality assurance	.073	.032	.179	2.281	.024

a. Dependent Variable: x6.Business Functional KSA

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.7737	3.1363	2.9487	.10832	160
Residual	-1.59129	1.08122	.02256	.57837	160
Std. Predicted Value	-1.615	1.732	.000	1.000	160
Std. Residual	-2.658	1.806	.038	.966	160

a. Dependent Variable: x6.Business Functional KSA

11.8.1.10 Production support

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12l.Production support[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x2.Software Development KSA

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.164[a]	.027	.021	.60998	2.045

a. Predictors: (Constant), A12l.Production support

b. Dependent Variable: x2.Software Development KSA

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.632	1	1.632	4.386	.038[a]
	Residual	59.160	159	.372		
	Total	60.792	160			

a. Predictors: (Constant), A12l.Production support

b. Dependent Variable: x2.Software Development KSA

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.861	.096		29.674	.000
	A12l.Production support	.063	.030	.164	2.094	.038

a. Dependent Variable: x2.Software Development KSA

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.8608	3.1738	3.0358	.10099	161
Residual	-1.92337	1.13924	.01361	.61424	161
Std. Predicted Value	-1.733	1.367	.000	1.000	161
Std. Residual	-3.153	1.868	.022	1.007	161

a. Dependent Variable: x2.Software Development KSA

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12I.Production support[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x6.Business Functional KSA

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.165[a]	.027	.021	.60013	1.955

a. Predictors: (Constant), A12I.Production support

b. Dependent Variable: x6.Business Functional KSA

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.607	1	1.607	4.462	.036[a]
	Residual	57.266	159	.360		
	Total	58.873	160			

a. Predictors: (Constant), A12I.Production support

b. Dependent Variable: x6.Business Functional KSA

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.775	.095		29.256	.000
	A12I.Production support	.062	.029	.165	2.112	.036

a. Dependent Variable: x6.Business Functional KSA

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.7750	3.0857	2.9487	.10022	161
Residual	-1.56143	1.22499	.03392	.57851	161
Std. Predicted Value	-1.733	1.367	.000	1.000	161
Std. Residual	-2.602	2.041	.057	.964	161

a. Dependent Variable: x6.Business Functional KSA

11.8.1.11 Preparation of manuals

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12j.Preparation of manuals[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x1.Org Culture

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.187[a]	.035	.029	.66852	2.038

a. Predictors: (Constant), A12j.Preparation of manuals

b. Dependent Variable: x1.Org Culture

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	2.539	1	2.539	5.682	.018[a]
	Residual	70.166	157	.447		
	Total	72.706	158			

a. Predictors: (Constant), A12j.Preparation of manuals

b. Dependent Variable: x1.Org Culture

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.244	.093		24.155	.000
	A12j.Preparation of manuals	.089	.037	.187	2.384	.018

a. Dependent Variable: x1.Org Culture

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.2443	2.6892	2.4261	.12677	160
Residual	-1.33326	1.75573	.00799	.66212	159
Std. Predicted Value	-1.435	2.075	.000	1.000	160
Std. Residual	-1.994	2.626	.012	.990	159

a. Dependent Variable: x1.Org Culture

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12j.Preparation of manuals[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x3.Management Skills

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.201[a]	.040	.034	.54457	1.766

a. Predictors: (Constant), A12j.Preparation of manuals

b. Dependent Variable: x3.Management Skills

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.965	1	1.965	6.625	.011[a]
	Residual	46.856	158	.297		
	Total	48.821	159			

a. Predictors: (Constant), A12j.Preparation of manuals

b. Dependent Variable: x3.Management Skills

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.722	.075		36.073	.000
	A12j.Preparation of manuals	.078	.030	.201	2.574	.011

a. Dependent Variable: x3.Management Skills

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.7215	3.1117	2.8810	.11116	160
Residual	-1.41018	1.27845	.00014	.54419	160
Std. Predicted Value	-1.435	2.075	.000	1.000	160
Std. Residual	-2.590	2.348	.000	.999	160

a. Dependent Variable: x3.Management Skills

11.8.1.12 Project management

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A12m.Project Management[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x3.Management Skills

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.184[a]	.034	.028	.54644	2.017

a. Predictors: (Constant), A12m.Project Management

b. Dependent Variable: x3.Management Skills

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.625	1	1.625	5.442	.021[a]
	Residual	46.582	156	.299		
	Total	48.206	157			

a. Predictors: (Constant), A12m.Project Management

b. Dependent Variable: x3.Management Skills

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.759	.068		40.464	.000
	A12m.Project Management	.065	.028	.184	2.333	.021

a. Dependent Variable: x3.Management Skills

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.7585	3.0811	2.8810	.10173	158
Residual	-1.40661	1.11245	-.01817	.53684	158
Std. Predicted Value	-1.204	1.967	.000	1.000	158
Std. Residual	-2.574	2.036	-.033	.982	158

a. Dependent Variable: x3.Management Skills

11.9 Total Professional Experience and knowledge requirements

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	Total Professional Experience[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x7.Problem solving abilities

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.186[a]	.034	.029	.55273	2.055

a. Predictors: (Constant), Total Professional Experience

b. Dependent Variable: x7.Problem solving abilities

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.810	1	1.810	5.925	.016[a]
	Residual	50.715	166	.306		
	Total	52.525	167			

a. Predictors: (Constant), Total Professional Experience

b. Dependent Variable: x7.Problem solving abilities

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	3.076	.072		42.788	.000
	Total Professional Experience	.014	.006	.186	2.434	.016

a. Dependent Variable: x7.Problem solving abilities

Residuals Statistics[a]

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	3.0766	3.5699	3.2173	.10411	170
Residual	-1.57756	.92230	-.01238	.55261	168
Std. Predicted Value	-1.352	3.386	.000	1.000	170
Std. Residual	-2.854	1.669	-.022	1.000	168

a. Dependent Variable: x7.Problem solving abilities

11.10 Experience within the project team and knowledge requirements

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	A7.Experience in Software Project team[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: x5.Communication and relationship skills

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.148[a]	.022	.016	.55363	1.792

a. Predictors: (Constant), A7.Experience in Software Project team

b. Dependent Variable: x5.Communication and relationship skills

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.210	1	1.210	3.949	.048[a]
	Residual	53.945	176	.307		
	Total	55.155	177			

a. Predictors: (Constant), A7.Experience in Software Project team

b. Dependent Variable: x5.Communication and relationship skills

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	3.174	.058		54.713	.000
	A7.Experience in Software Project team	.031	.016	.148	1.987	.048

a. Dependent Variable: x5.Communication and relationship skills

11.11 *Factor correlation matrix*

Correlations

		x1.Org Culture	x2.Project Software Dev. KSA	x3. ManagementSkills	x4. Communication and relationship skills	x5.Heuristics knowledge	x6.Business Functional KSA	x7.Problem Solving with help
x1.Org Culture	Pearson Correlation	1	.410**	.610**	.486**	.529**	.408**	.381**
	Sig. (2-tailed)		.000	.000	.000	.000	.000	.000
	N	177	177	177	177	176	177	175
x2.Project Software Dev. KSA	Pearson Correlation	.410**	1	.357**	.348**	.421**	.457**	.460**
	Sig. (2-tailed)	.000		.000	.000	.000	.000	.000
	N	177	178	178	178	176	178	176
x3.ManagementSkills	Pearson Correlation	.610**	.357**	1	.625**	.546**	.467**	.400**
	Sig. (2-tailed)	.000	.000		.000	.000	.000	.000
	N	177	178	178	178	176	178	176
x4.Communication and relationship skills	Pearson Correlation	.486**	.348**	.625**	1	.441**	.392**	.528**
	Sig. (2-tailed)	.000	.000	.000		.000	.000	.000
	N	177	178	178	178	176	178	176
x5.Heuristics knowledge	Pearson Correlation	.529**	.421**	.546**	.441**	1	.398**	.434**
	Sig. (2-tailed)	.000	.000	.000	.000		.000	.000
	N	176	176	176	176	176	176	174
x6.Business Functional KSA	Pearson Correlation	.408**	.457**	.467**	.392**	.398**	1	.430**
	Sig. (2-tailed)	.000	.000	.000	.000	.000		.000
	N	177	178	178	178	176	178	176
x7.Problem Solving with help	Pearson Correlation	.381**	.460**	.400**	.528**	.434**	.430**	1
	Sig. (2-tailed)	.000	.000	.000	.000	.000	.000	
	N	175	176	176	176	174	176	176

**. Correlation is significant at the 0.01 level (2-tailed).

11.12 Additional statistics

The SPSS data file is enclosed in a CD along with the thesis. Additional statistics can be generated out of the SPSS data file.

11.13 SE Body of Knowledge (SWEBOK)

IEEE published a guide to SE Body of Knowledge in 2004. This work by IEEE is an attempt to define the core body of knowledge for SE so that the field of SE discipline can be considered as a legitimate profession and an engineering discipline. The guide to SWEBOK was developed by a large panel of experts from industry and academia. The guide went through three phases: a prototype phase called straw-man, finalisation of a trial version in the phase termed stone-man and final revision and approval in the iron-man phase. The whole project to develop guide to SWEBOK was carried out with underlying principles that were based on transparency and a consensus approach.

The following objectives were set out for SWEBOK (IEEE, 2004):

- characterize the contents of the SE Body of Knowledge;
- provide a topical access to the SE Body of Knowledge;
- promote a consistent view of SE worldwide;
- clarify the place of, and set the boundary of, SE with respect to other disciplines such as computer science, project management, computer engineering and mathematics;
- Provide a foundation for curriculum development and individual certification and licensing material.

The SWEBOK only seeks to identify the core body of knowledge, referred to as generally accepted body of knowledge. It does not aim to document specialised knowledge or advanced and research based knowledge for the SE field.

The guide to SWEBOK uses a hierarchical structure to classify the various knowledge types into knowledge areas, sub-areas, topics and sub-topic. The following hierarchical charts (see Figure 11.1 and Figure 11.2) show the structure of guided to SWEBOK.

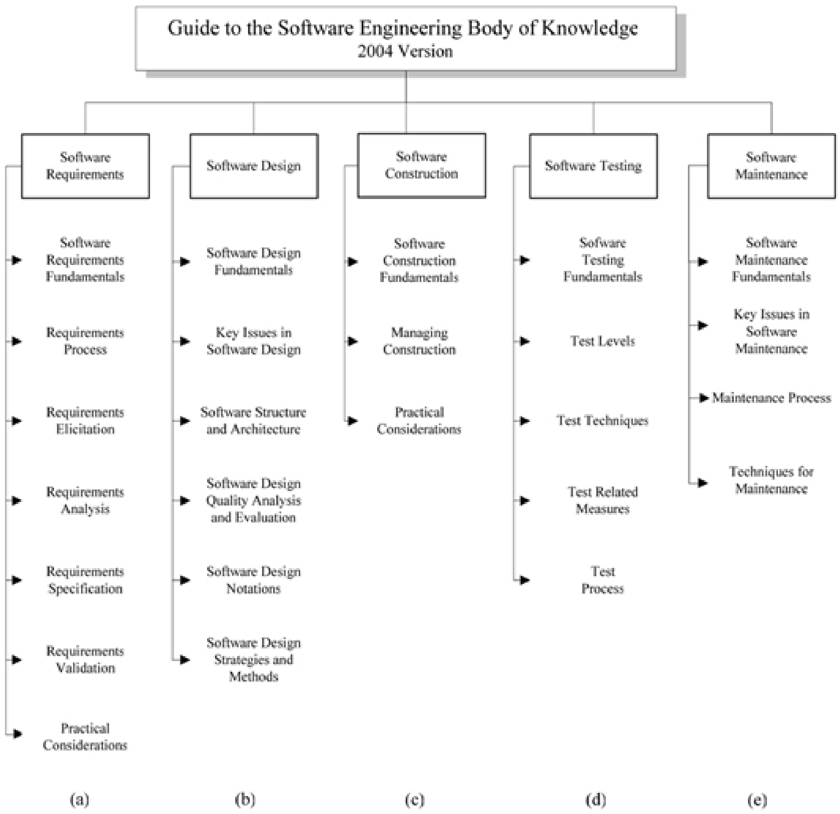

Figure 11.1. Guide to SE Body of Knowledge – Part A (IEEE, 2004)

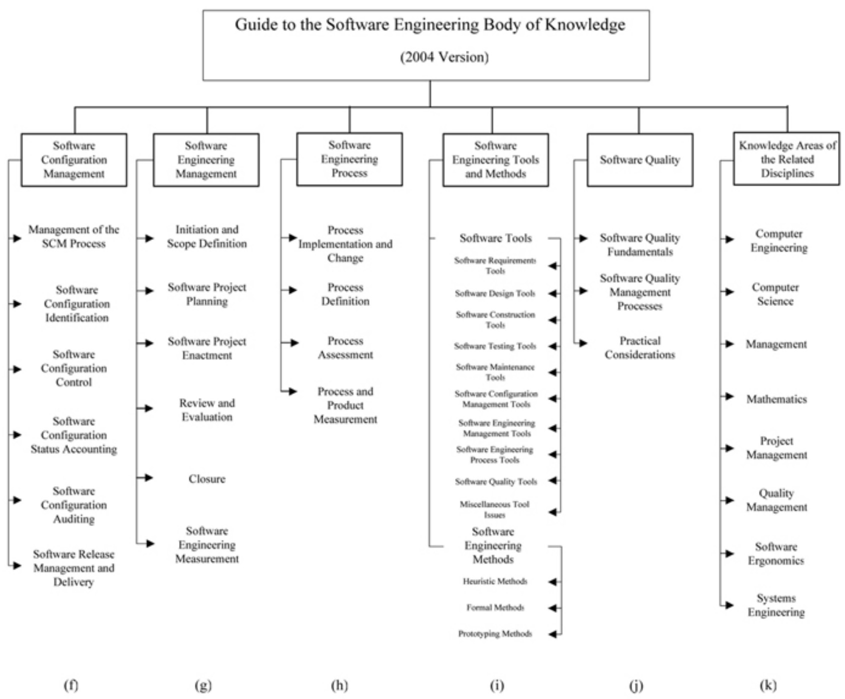

Figure 11.2. Guide to SE Body of Knowledge – Part B (IEEE, 2004)

11.14 People Capability Maturity Model (P-CMM)

People-CMM proposed by Software Engineering Institute (SEI) addresses the workforce practices in the organisation. People CMM provides a framework that integrates the people practices with technology and processes (Kulpa, 2007). A five level step by step process maturity framework has been proposed by SEI. The diagram in Figure 11.3 shows the architecture of People-CMM. People-CMM, based on best practices in HRM and KM fields, helps organisations characterize the maturity of their workforce practices, establish a program of continuous workforce development, set priorities for improvement actions, integrate workforce development with process improvement, and establish a culture of excellence (Curtis et al., 2003).

Levels	Process Area Threads			
	Developing Competency	Building Workgroups and Culture	Motivating and Managing Performance	Shaping the Workforce
5 Optimizing	Continuous Capability Improvement		Organizational Performance Alignment	Continuous Workforce Innovation
4 Predictable	Mentoring Competency Based Assets	Competency Integration Empowered Workgroups	Quantitative Performance Management	Organizational Capability Management
3 Defined	Competency Development Competency Analysis	Workgroup Development Participatory Culture	Competency Based Practices Career Development	Workforce Planning
2 Managed	Training and Development	Communication and Coordination	Compensation Performance Management Work Environment	Staffing

Figure 11.3. Architecture of People-CMM (Curtis et al., 2003)

11.15 Profile of respondents

Other role	Number of respondents
A1-IT-security-expert	1
A2-Systems-developer	1
A3-Release-manager	1
A4-web editor	1
A5-Tech support	2
A6-System-Admin	1
A7-database-analyst	1
A8-DBA	3
Total	11

Table 11.2. Profile of respondents who marked 'other' roles

Short identification#	Business analyst	Programmer Analyst	Project Manager	Architect	Tester	Designer	No. of respondents
B81-BA-PA-Test-Arch	X			X	X		1
B82-BAPrg-Arch-De-PM	X						1
B83-PA-Architect		X		X			1
B84-PM-Architect			X	X			1
B91-BA-PM	X		X				1
B92-PA-Arch-Tester		X		X	X	X	1
B93-ProgAnaly-Design		X					2
B94-ProgAna-PM		X	X				1
Total							9

Table 11.3. Count of respondents who marked multiple roles#

- BA – Business Analyst, PA – Programmer Analyst, PM – Project Manager, De – Designer, Design – Designer, Arch – Architect,

City	Number of respondents
Bangalore	1
Boulder	1
Broomfield	7
Canberra	113
Chennai	25
Delhi	1
Horsham	1
Hyderabad	1
Kansas	1
London	3
Omaha	8
Philadelphia	1
Singapore	4
St. Louis	1
Sydney	5
Trenton	5
Total	178

Table 11.4. City based profile of respondents

11.16 Summary of literature

Table 11.5 summarise the literature review based on various categorisations of knowledge and skill requirements. Note that the column C7 in Table 11.5, denotes if the derived knowledge and skill requirements have been founded on sound fundamental theories.

No. (C0)	References (C1)	Research method (C2)	Sample size/number of case units (C3)	Roles (C4)	Source of data (C5)	Main purpose (C6)	Theory based (C7)	Knowledge, Skill, Abilities (C8)
1.	Henry et al. (1974 cited in Benbasat et al., 1980)	Survey		IS professionals	IS professionals	Curriculum refinement	No	people, organisation, society, systems, computers and models
2.	Benbasat et al.(1980)	Survey	35 organisations	IS managers and Systems analysts	IS Managers and systems analysts	Curriculum refinement	No	Generalist and Specialist skills
3.	McCubbray and Scudder (1988)	Interviews	Not specified	Systems analyst	IS practitioners	Identification of skills	No	People, organisation, society, systems, computers and models.
4.	Cash et al. (2004)	Case study	15		IS professionals	Understand skills	No	Technical, business, conceptual and relationship skills.
5.	Curtis et al. (1988)	Field study	17 large projects	Designers	Members of software design teams	Gain an ecological perspective of software design process	No	Communication skills, social network knowledge and application domain knowledge
6.	Herbsleb and Kuwana (1993)	Meeting minutes and video taping of design meetings	38 design meeting minutes, three video taped meetings at two large organisations	Designers	Members of software design teams	To inform design of technology support and new procedural methods for software design	No	Application system knowledge such as user scenarios, application domain knowledge and knowledge of requirements
7.	Walz et al. (1993)	Longitudinal case study	One software project	Designers	Members of design team	Better SE management of large software projects	No	Application domain knowledge Better Knowledge Management required in design teams.
8.	Zmud, (1983 pp.257-258)	Published in a book	Not applicable	Both end users and IS professionals	Not applicable	System implementation	No	Organisational overview, organisational skills, organisational unit, general IS, technical skills and IS product
9.	Nelson (1991)	Survey	275 participants	Both end users and IS professionals	Both end users and IS professionals	To identify educational needs	No	Zmud's (1983) categorisation as above.
10.	Nansi and Bennett (1998)	Survey	54 responses	IS managers	IS Managers	Individual skills	No	Zmud's (1983) categorisation as above.
11.	Lee et al. (1995)	Survey	52 IS managers, 15 user managers and 30 IS consultants	All IS professionals	IS managers, user managers IS consultants	Macro level HRM skills identification for Boston area and better curriculum design	No	Business functional skills, Interpersonal management, Technical skills, Technology management skills.
12.	Sawyer et al (1998)	Case study	140 respondents from a single company	All IS professionals	IS staff of the company	Identify company's skill needs	No	The survey used the same classification as Lee, Trauth et al. (1995) The qualitative study resulted in the following skills : Core set of IT skills: Need for interpersonal skills Need for project management skills IT skills useful in organisation Combination of skills to be a problem solver

No. (C0)	References (C1)	Research method (C2)	Sample size/number of case units (C3)	Roles (C4)	Source of data (C5)	Main purpose (C6)	Theory based (C7)	Knowledge, Skill, Abilities (C8)
13.	Cheney and Lyons (1980) and Cheney (1988)	Survey		Programmer, systems analyst and project managers	IS Managers	Macro level Human resource point of view	No	No formal classification. Technical and Managerial skills
14.	Leitheiser (1992)	Survey		Developers	IS Managers	Macro level Human resource point of view	No	Analysis and design, programming, interpersonal, business, environment (ex: PC, mainframe), language (example: COBOL, Job Control Language (JCL)) and application
15.	Monin and Dewe (1994)	Survey		All roles	IS managers	Macro level Human resource point of view	No	Generalist-Managerial Generalist-Technical Specialist-Managerial Specialist-Technical
16.	Becker et al. (1997)	Survey	11 firms, 57 responses	Managerial, technical, operational and Technical support	Not specified	To identify soft skills	No	Importance of communication skills to all IS jobs
17.	Misic, (1996)	Survey	134 responses with 38.8% response rate	Systems analysts	Systems analysts	Identify most important skills to develop today's Information systems	No	Analytical skills, technical skills, communication skills and interpersonal skills.
18.	Bailey and Stefaniak (2001)	Survey		Programmers	Programmers	Macro level HRM	No	Technical skills, soft skills and business concept skills
19.	Todd et al. (1995)	Content analysis of newspaper advertisements		Programmers, systems analysts and project managers	Not applicable.	Understand skill requirements for recruitment	No	Technical knowledge/skills, Business knowledge/skills and Systems knowledge
20.	Nansi and Bennet (1998)	As above		IS managers	IS Managers	As above	No	Zmud's (1983) categorisation
21.	Wade and Parent (2001)	As above		Webmasters	Webmasters	Build theory	No	Technical skills and organisational skills (including communication skills, general management skills, ability to work in teams, project management skills)
22.	Chan (2003)	Content analysis of ValueSAP documentation		Consultant, users, training team, technical team and steering committee	Not applicable	knowledge management	No	Project management knowledge, business knowledge, product-specific knowledge, company-specific knowledge and technical knowledge
23.	The Open group (2007b)	Not applicable	Not applicable	Architect	TOGAF framework	Industry Body of Knowledge	No	Generic Skills, Business Skills etc. Enterprise Architecture Skills:, Program or Project Management Skills, IT General Knowledge Skills, Technical IT Skills, Legal Environment
24.	Schwalbe (2002)	Not applicable	Not applicable	Project Manager		Industry Body of Knowledge	No	Management skills, organisational skills, Use of technology.

Table 11.5. Summary of literature

This last page is left intentionally blank