

Tacit Knowledge in a Software Development Project

Hanna Dreyer

BSc, MSc;

A thesis submitted to the University of Gloucestershire in accordance with the requirements of the degree of PhD in the Faculty of Business, Computing and Applied Sciences.

Supervisors: Dr. Robin Bown

Dr. Martin George Wynn

Submission: April 2018

I declare that the work in this thesis was carried out in accordance with the regulations of the University of Gloucestershire and is original except where indicated by specific reference in the text. No part of the thesis has been submitted as part of any other academic award. The thesis has not been presented to any other education institution in the United Kingdom or overseas.

Any views expressed in the thesis are those of the author and in no way represent those of the University.

Signed:

Student Number:

Date:

Abstract

Knowledge, and tacit knowledge in particular, is a key component of a successful software development project. Although the importance of tacit knowledge has been widely acknowledged by researchers, few have undertaken an in-depth exploratory investigation of its use within the software development process. Its intangible nature makes it difficult to conceptualize and is therefore challenging to investigate. This study explores tacit and explicit knowledge in software development project meetings, focusing on their acquisition and sharing within the team as well as on an individual basis. The interplay between individual and group tacit knowledge is particularly interesting when observing a project team over time. By analysing knowledge sources, a knowledge development process emerges, which sheds light upon the growth and exchange of tacit knowledge within the team.

The investigation is of a UK based HR software development project observed through participant observation over a three-month period, involving the software development organization, a human resource consultancy and an organization focusing on archaeology. The three companies worked together to complete a tailored software package for the organization, complementing each other in their expertise within each of their fields. The centre for knowledge exchange within the project was the weekly meetings conducted through face-to-face conversations and conference calls, exposing tacit knowledge at its point of creation. The research concludes with a model representing the process of tacit knowledge development and exchange within a software development project from both an individual and team based view. The existence of a common dynamic environment with qualified individuals exchanging tacit knowledge and making decisions demonstrates the importance of face-to-face conversations. The model helps develop an understanding of the points of tacit knowledge acquisition, sharing and creation, as well as their usage by everyone. The findings are used to make recommendations concerning the management of knowledge exchange within software development project teams as well as highlighting possible areas for future research in this field.

Acknowledgements

This journey would not have been possible without the support and encouragement of a great many people over the past years. The contributions made are sincerely appreciated and acknowledged. In particular, I would like to thank the following people:

First, I would like to thank my supervisors, Dr. Robin Bown and Dr. Martin George Wynn without whom the completion of the thesis would not have been possible.

A special thank you to my parents, Brigitte and Joerg Dreyer. Thank you for your love and support throughout the years, without you I would have never gone on this journey nor finished it.

To my friends who have stood by me and showed their support throughout the way, thank you.

Table of Contents

Chapter 1: Introduction.....	19
1.1 Background	19
1.2 Acquiring and Sharing Knowledge in Software Projects	20
1.3 Research Objectives and Questions	23
1.4 Summary and thesis structure	25
Chapter 2: Theoretical Framework	27
2.1 Introduction	27
2.2 Knowledge Management	28
2.2.1 Knowledge Assets	28
2.2.2 Knowledge as Human Capital	31
2.3 Tacit Knowledge.....	41
2.3.1 Elements of Tacit Knowledge	46
2.3.2 Tacit Knowledge in an Organization	48
2.3.3 Knowledge Transfer in an Organization	52
2.3.4 A Dynamic Environment to Create and Share Tacit Knowledge ..	55
2.3.5 Channels for the Exchange and Sharing of Tacit Knowledge	57
2.4 Analysing Tacit Knowledge from an Individual Perspective	58
2.5 Team Tacit Knowledge	65
2.6 Decision Making and Tacit Knowledge	71
2.7 Conclusions.....	73
Chapter 3: Methodology	79
3.1 Introduction	79
3.2 Research Approach	80
3.3 Methodological Approach	82
3.3.1 Grounded Theory.....	86
3.3.2 Coding	87
3.3.3 Modelling and Displaying Concepts.....	88
3.3.4 Ethnographic Decision Making	88
3.4 Organisations, People and the Project	89
3.4.1 The Companies and People Involved	89
3.4.2 The Project	91
3.5 Data Collection and Analysis.....	92

3.5.1 Collection	92
3.5.2 Analysis	94
3.6 Researcher's Perspective	101
3.7 Ethical Considerations	102
3.8 Summary.....	103
Chapter 4: Concepts, Theories and Results	105
4.1 Introduction	105
4.2 Results from the 3 Research Cycles	105
4.2.1 Research Cycle 1	105
4.2.2 Research Cycle 2	107
4.2.3 Research Cycle 3	112
4.3 Summary.....	116
Chapter 5: Findings & Analysis	119
5.1 Introduction	119
5.2 Analysing the Data.....	121
5.2.1 Finance	121
5.2.2 Human Resource Tools – 360-degree feedback	145
5.2.3 Time (Working Patterns, Leave)	150
5.2.4 Recruitment	158
5.2.5 Employee Centre	180
5.2.6 Disciplinary and Grievances	192
5.2.7 Project Plan	197
5.2.8 Summary	202
5.3 Decision making and tacit knowledge	203
5.2 Summary.....	226
Chapter 6: Derivation of the Framework	229
6.1 Introduction	229
6.2 Tacit Knowledge Triggers	230
6.2.1 Visual Triggers.....	231
6.2.2 Conversational Triggers.....	232
6.2.3 Constructive Learning Triggers.....	233
6.2.4 Anticipation Triggers	234
6.2.5 Recall Triggers.....	235

6.3 The Model – Recognizing and Harnessing Tacit Knowledge	237
6.3.1 Combining the Models in a Meeting Context	237
6.3.2 The Model and its relation to the exchange of tacit knowledge ..	245
6.4 Summary.....	258
Chapter 7: Results and Conclusion.....	259
7.1 Introduction	259
7.2 Key Conclusions	260
7.3 Methodological Issues and Limitations of the Research	262
7.4 Recommendations and Future Research	264
7.5 Contribution and Implications	265
References	269
Appendix	286
Evaluation 1. Cycle Excel Sheet Example	287
Evaluation 2. Cycle Excel Sheet Example	288
Evaluation 3. Cycle Excel Sheet Example	289

List of Tables

Table 1 - Epitomes, Clarke (2010)	61
Table 2 - Tacit Elements, Clarke (2010).....	63
Table 3 - Example Table Tacit Knowledge Data	98
Table 4 - Example Table Tacit Knowledge Triggers Data	98
Table 5 - Comparison Theories of Nonaka, Ryan and Clarke.....	111
Table 6 - Recorded Meetings Table	120
Table 7 - Extract 1 Results Tacit Knowledge	122
Table 8 - Extract 1 Results Tacit Knowledge Triggers	122
Table 9 - Extract 2 Results Tacit Knowledge	124
Table 10 - Extract 2 Results Tacit Knowledge Triggers	124
Table 11 - Extract 3 Results Tacit Knowledge	125
Table 12 - Extract 3 Results Tacit Knowledge Triggers	125
Table 13 - Extract 4 Results Tacit Knowledge	126
Table 14 - Extract 4 Results Tacit Knowledge Triggers	127
Table 15 - Extract 5 Results Tacit Knowledge	128
Table 16 - Extract 5 Results Tacit Knowledge Triggers	128
Table 17 - Extract 6 Results Tacit Knowledge	129
Table 18 - Extract 6 Results Tacit Knowledge Triggers	129
Table 19 - Extract 7 Results Tacit Knowledge	131
Table 20 - Extract 7 Results Tacit Knowledge Triggers	131
Table 21 - Extract 8 Results Tacit Knowledge	133
Table 22 - Extract 8 Results Tacit Knowledge Triggers	133
Table 23 - Extract 9 Results Tacit Knowledge	135
Table 24 Extract 9 Results Tacit Knowledge Triggers	135
Table 25 - Extract 10 Results Tacit Knowledge	136
Table 26 - Extract 10 Results Tacit Knowledge Triggers	136
Table 27 - Extract 11 Results Tacit Knowledge	137
Table 28 - Extract 11 Results Tacit Knowledge Triggers	137
Table 29 - Extract 12 Results Tacit Knowledge	138

Table 30 -Extract 12 Results Tacit Knowledge Triggers	138
Table 31 - Extract 13 Results Tacit Knowledge	140
Table 32 - Extract 13 Results Tacit Knowledge Triggers	141
Table 33 - Extract 14 Results Tacit Knowledge	142
Table 34 - Extract 14 Results Tacit Knowledge Triggers	142
Table 35 - Extract 15 Results Tacit Knowledge	143
Table 36 - Extract 15 Results Tacit Knowledge Triggers	143
Table 37 - Extract 16 Results Tacit Knowledge	144
Table 38 - Extract 16 Results Tacit Knowledge Triggers	144
Table 39 - Extract 17 Results Tacit Knowledge	146
Table 40 - Extract 17 Results Tacit Knowledge Triggers	147
Table 41 - Extract 18 Results Tacit Knowledge	149
Table 42 - Extract 18 Results Tacit Knowledge Triggers	149
Table 43 - Extract 19 Results Tacit Knowledge	153
Table 44 - Extract 19 Results Tacit Knowledge Triggers	153
Table 45 - Extract 20 Results Tacit Knowledge	155
Table 46 - Extract 20 Results Tacit Knowledge Triggers	155
Table 47 - Extract 21 Results Tacit Knowledge	156
Table 48 - Extract 22 Results Tacit Knowledge Triggers	156
Table 49 - Extract 22 Results Tacit Knowledge	158
Table 50 - Extract 22 Results Tacit Knowledge Triggers	158
Table 51 - Extract 23 Results Tacit Knowledge	160
Table 52 - Extract 23 Results Tacit Knowledge Triggers	160
Table 53 - Extract 24 Results Tacit Knowledge	161
Table 54 - Extract 24 Results Tacit Knowledge Triggers	161
Table 55 - Extract 25 Results Tacit Knowledge	162
Table 56 - Extract 25 Results Tacit Knowledge Triggers	163
Table 57 - Extract 26 Results Tacit Knowledge	164
Table 58 - Extract 26 Results Tacit Knowledge Triggers	164
Table 59 - Extract 27 Results Tacit Knowledge	166

Table 60 - Extract 27 Results Tacit Knowledge Triggers	166
Table 61 - Extract 28 Results Tacit Knowledge	167
Table 62 - Extract 28 Results Tacit Knowledge Triggers	167
Table 63 - Extract 29 Results Tacit Knowledge	169
Table 64 - Extract 29 Results Tacit Knowledge Triggers	169
Table 65 - Extract 30 Results Tacit Knowledge	173
Table 66 - Extract 30 Results Tacit Knowledge Triggers	173
Table 67 - Extract 31 Results Tacit Knowledge	175
Table 68 - Extract 31 Results Tacit Knowledge Triggers	175
Table 69 - Extract 32 Results Tacit Knowledge	176
Table 70 - Extract 32 Results Tacit Knowledge Triggers	176
Table 71 - Extract 33 Results Tacit Knowledge	177
Table 72 - Extract 33 Results Tacit Knowledge Triggers	177
Table 73 - Extract 34 Results Tacit Knowledge	178
Table 74 - Extract 34 Results Tacit Knowledge Triggers	178
Table 75 - Extract 35 Results Tacit Knowledge	179
Table 76 - Extract 35 Results Tacit Knowledge Triggers	179
Table 77 - Extract 36 Results Tacit Knowledge	181
Table 78 - Extract 36 Results Tacit Knowledge Triggers	181
Table 79 - Extract 37 Results Tacit Knowledge	186
Table 80 - Extract 37 Results Tacit Knowledge Triggers	186
Table 81 - Extract 38 Results Tacit Knowledge	188
Table 82 - Extract 38 Results Tacit Knowledge Triggers	188
Table 83 - Extract 39 Results Tacit Knowledge	189
Table 84 - Extract 39 Results Tacit Knowledge Triggers	189
Table 85 - Extract 40 Results Tacit Knowledge	190
Table 86 - Extract 40 Results Tacit Knowledge Triggers	191
Table 87 - Extract 41 Results Tacit Knowledge	192
Table 88 - Extract 41 Results Tacit Knowledge Triggers	192
Table 89 - Extract 42 Results Tacit Knowledge	195

Table 90 - Extract 42 Results Tacit Knowledge Triggers	196
Table 91 - Extract 43 Results Tacit Knowledge	197
Table 92 - Extract 43 Results Tacit Knowledge Triggers	197
Table 93 - Extract 44 Results Tacit Knowledge	199
Table 94 - Extract 44 Results Tacit Knowledge Triggers	199
Table 95 - Extract 45 Results Tacit Knowledge	201
Table 96 - Extract 45 Results Tacit Knowledge Triggers	202
Table 97 - Nonaka and Ryan comparison	241
Table 98 - Nonaka, Ryan, Clarke Comparison.....	243

List of Figures

Figure 1 - Spiral of Knowledge (Nonaka and Teece, 2001)	58
Figure 2 - Spiral of Knowledge Creation (Nonaka and Takeuchi, 1995)	59
Figure 3 - Explicit and Tacit Knowledge as Two Dimensions of Knowledge (McAdam et al., 2007)	60
Figure 4 - Epitomes of Tacit Knowledge (McAdam et al., 2007).....	62
Figure 5 - The Reflective Process (Clarke, 2010)	63
Figure 6 - Triggered Response Loop (Clarke, 2010).....	64
Figure 7 - The Tacit Knowledge Transfer Model (Clarke, 2010)	64
Figure 8 - The Tacit Knowledge Spectrum (Clarke, 2010)	65
Figure 9 - Theoretical Model for the Acquisition and Sharing of Tacit Knowledge in Teams (Ryan, 2013).	70
Figure 10 - Natural Decision Making Model (Klein, 1999).....	72
Figure 11 - Companies involved in the Project and their needs	106
Figure 12 - Number of Tacit Knowledge found in each Category	113
Figure 13 – Number of Tacit Knowledge Triggers found in Data	115
Figure 14 – Knowledge Creation and its Relationship to Trigger Points	116
Figure 15 - Recognition-primed decision model (after Klein et al., 1989) ..	203
Figure 16 - Extract 46 Decision Making	206
Figure 17 - Extract 47 Decision Making	209
Figure 18 - Extract 48 Decision Making	212
Figure 19 - Extract 49 Decision Making	214
Figure 20 - Extract 50 Recall Decisions	217
Figure 21 - Extract 51 Recall Decisions	221
Figure 22 - Extract 52 Recall Decisions	224
Figure 23 - Extract 53 Recall Decisions	226
Figure 24 - ‘Ba’ (Nonaka and Teece, 2001)	239
Figure 25 - ‘Ba’ and SEI (Nonaka and Teece, 2001)	239
Figure 26 - Theoretical Model for the Acquisition and Sharing of Tacit.....	240
Figure 27 - Combination of Theories Nonaka and Teece (2010) and Ryan (2013)	241

Figure 28 - Tacit Knowledge Spectrum (Clarke, 2010)	242
Figure 29 - Internalization Process.....	244
Figure 30 - Socialization, Externalization and Internalization	244
Figure 31 - Knowledge in and output to internalization	250
Figure 32 - Tacit Knowledge Model Process.....	252
Figure 33 - Individual and Group Tacit Knowledge Model	256

Chapter 1: Introduction

1.1 Background

Communication is one of the main reasons for project failure according to the project management institute (2008). When assessing project team's miscommunication, it often arises due to meetings and other forms of integration not being efficient. Bozarth (2014) encourages sharing your work through working out loud and therefore allowing others to learn while a person is performing a specific task. Creating a safe space for knowledge exchange should therefore be at the centre of a project. It is the aim of the thesis to analyse project teams and their knowledge exchange within meetings.

Project teams consist of different project members, each of which are experts within their field. These are selected to perform a specific task within the project, where each task is then assembled to create a common goal. It is therefore crucial that each team member works first, efficiently and secondly, properly communicates their knowledge to the team to allow the project to assemble properly. The knowledge surrounding the project is therefore quite vast and in different places.

A software development project greatly relies on the expertise of different players. The creation of such a software is usually quite fast and changes can be made quickly due to its intangible technical nature. This thesis will focus on the development of a human resource software. Three main groups develop the software: the customer, the software development organization and a human resource consultancy. Each player allows different knowledge and expertise to be communicated and utilized. Each project member uses their expert knowledge to create the software and shares this with the team.

Tacit knowledge is one of the most difficult forms of knowledge to share and acquire during a project due to its intangible nature. Tacit knowledge is at the core of a knowledge based society and its exchange is still of great interest to researchers. How tacit knowledge is exchanged and used within the different project teams plays a vital role in project success. Banacu

(2013) stresses the importance of tacit knowledge transfer due to companies needing it to obtain a competitive advantage. Throughout this thesis, the aim will be to analyse a project team's tacit knowledge exchange within a software development meeting environment.

This chapter will introduce knowledge sharing and acquisition, followed by the research objectives and a road map of the thesis.

1.2 Acquiring and Sharing Knowledge in Software Projects

Tacit and explicit knowledge are at the core of a software development project. It has been the focus of recent studies to examine the relationship between tacit and explicit knowledge in software development. The connection between the tacit and the explicit knowledge easily determines whether or not a software project achieves its full potential. Most people are familiar with explicit knowledge, which generally entails written instructions, guidelines, protocols and step-by-step processes. Information or knowledge can be conveyed through written or verbal instructions and the receiver transforms this into a comprehensive message, if possible. Tacit knowledge unlike explicit knowledge, is more difficult to comprehend and transmit. It operates at the innate or instinctive level, and is, therefore, hard to convey and even more difficult to understand. It rests at the level of the subconscious and involves a mental scheme which is mostly taken for granted and is rarely deconstructed. Due to its instinctive nature, tacit knowledge is not thought about when produced. According to Winter (2016) there is a path to acquiring tacit knowledge without a transfer. By using a trial and error approach through a teacher tacit knowledge can be acquired. It is a slower process and takes devotion to receive and give feedback. Given the immense potential of tacit knowledge, the understanding and articulation of it may be one of the most significant things an organisation can achieve.

The importance of the transformation of tacit knowledge into explicit team knowledge in software development projects is critical. Leonard and Swap (2014) argue that when an employee leaves an organization it loses

critical tacit knowledge which it cannot replace. An organisation must therefore be able to create excellence and expertise within its time limited project teams, in order to retain the knowledge from one project to the next. It may be argued that not everybody will be an expert, or naturally gifted at something; this is where the tacit knowledge of the performers needs to be transferred for those which do not have such a knowledge foundation. Evaluating the subject suggests that effective social networks, mentoring and modelling practices are critical to achieve success. With proper practice, little by little, that which is difficult to apprehend - or difficult to capture in words - can be understood by those who have a greater capacity for learning and accomplishment than they might envision. Before assessing the literature, it seems, software development projects clearly have to feature close feedback, assessing intimate ties between colleagues and a recognition that different people learn optimally via the use of different models. People are capable of building up their own tacit knowledge foundations and use those to gradually foster a growing base of explicit knowledge within a project.

McAfee (2003) writes that checklists alone cannot lead to successful IT projects and to fruitful implementation. Leadership is a subtle craft and a subtle art, and proper knowledge transfer does not easily lend itself to simply interpreting checklists. This is especially so when there are subtle gradations with regards to when a project is truly successful – or may simply be done “mostly” right. So many pitfalls – inertia, mis-specification, resistance, misuse and non-use – can cause a negative outcome for any IT initiative. Small missed steps in preparation, planning, leadership style, and even timing can lead to serious mishaps. In the end, even the most detailed and prescriptive projects can quickly run aground (McAfee, 2003). This is the essence of why tacit knowledge is so important, and why communicating tacit knowledge to the fullest extent possible, can be so beneficial: there are decisions which have to be made, adaptations which must occur, that will involve a bedrock of knowledge that is not likely to be found in any checklist. If that knowledge, which is difficult to articulate, can be shared, some of the loopholes cited above can be closed. Moreover, even if it is not possible to close every loophole, the more an organization is able to make tacit knowledge

comprehensible, the more it will be able to create orientation programs and training programs that teach project workers and all employees to develop the right habits of mind, the right competencies, and the necessary “read and react” skills to meet evolving circumstances properly; but it all starts with tacit knowledge and how tacit knowledge is utilized and understood.

Tacit knowledge is a medley of competencies and experiential recollections to which very little thought is given. It is something that is done reflexively and instinctively and very few of us tick off any boxes as we do it. Polanyi (1998) argues that tacit knowledge is the root of all knowledge and is knowledge in its rawest form: it must subsequently be turned into explicit, articulated knowledge for knowledge to achieve its full potential. Since this sort of knowledge is instinctive, derived from experience and innate competencies, it is probably best articulated through modelling and constant formative assessment. Put another way, tacit knowledge is the one form of knowledge that almost everyone must see if they wish to learn it well; merely reading about it, or seeing the descriptive words on a page, will be insufficient for permitting true transfer. The practices of transformation must include modelling, a cautiously constructivist approach to learning, interaction, and a democratic workplace that allows ideas to run freely.

In general, tacit knowledge and software development projects have a complicated but, potentially, fruitful relationship. Software development projects are, in most cases, always in a state of flux. Therefore, knowledge must be in a state of flux, too, and the mind must be open to all possibilities. Tacit knowledge is better adapted to this sort of endeavour than explicit knowledge, but tacit knowledge is also something that is difficult to articulate to others. The key, therefore, is to create an internal culture that does far more than make up a series of explicit protocols for how to get something done: the culture must, quite to the contrary, produce a space defined by sharing and caring – and by finding a wide array of means through which knowledge can be passed from one party to another. Finding the best means of optimizing and exploiting tacit knowledge in software development projects is the overall aim of this research.

1.3 Research Objectives and Questions

The aim of the research is to uncover tacit and explicit knowledge in software development projects. Focusing on tacit knowledge, the interplay between individual and group tacit knowledge is at the core of the research. Assessing a project in a software development environment over time allows tacit knowledge to flourish in an environment which is fast paced and where knowledge can be used as well as applied quickly. In the following section the research questions will be presented and explained.

Research Question 1:

What is the current understanding of knowledge exchange in software development projects?

To understand tacit knowledge, first an assessment of literature will be presented. The aim of the first research question is to understand tacit and explicit knowledge exchange in an IT software development project. To achieve this, the focus lies in tacit and explicit knowledge within an organization and the role it plays. Understanding knowledge and its importance within an organization helps understand the ties it has to a larger community. Drilling down in the topic, individual as well as group tacit knowledge will give a more detailed view on the impact tacit knowledge has. Playing in a software development project environment, tacit knowledge and its relation to software development and projects will be addressed. The literature is set to help build concepts and theories around the following three research questions.

Research Question 2:

How can tacit and explicit knowledge be recognised and evaluated in software development projects?

The assessment of how tacit and explicit knowledge are exchanged in a software development project is the second research question. The question is aimed to use concepts and theories gained from the previous research question and allows the combination of data and theories. The

methodology demonstrates how the data was collected and analysed, taking concepts and theories into account. Here, the flow of knowledge is at the core of the questions. Answering how tacit and explicit knowledge are exchanged in the software development project allows a deep investigation of the data and builds context between theory and data. This will be the focus of chapter 5, data analysis. The evaluated data can then be used to assess the next research question.

Research Question 3:

To what extent does non-communicated tacit and explicit knowledge amongst team members influence the project and its acceptance?

Decision making and tacit knowledge are the main topics of the following research question. The evolution of tacit and explicit knowledge in a software development project and its effect on individual and group decision making is the next research question. Using the gathered data, extracts of decisions are filtered out and assessed within chapters 5 and 6. These allow an assessment of tacit knowledge exchange and the usage of gained tacit knowledge during the project in relation to decision making. Demonstrating the importance of expert knowledge and using knowledge gained throughout the project, the question aims to show how tacit knowledge is used within the group by individuals.

Research Question 4:

Can tacit and explicit knowledge be better harnessed through the development of a conceptual model for use in software development projects?

The final research question is whether tacit and explicit knowledge can be recognized and harnessed by a conceptual model in a software development project. This question pulls together the theories and data analysed to visually demonstrate the flow of tacit knowledge within a software development project. The focus of the developed model is the interplay between individual and group tacit knowledge as well as the moments tacit knowledge is triggered. The model aims to help project

managers create a free space where a project can be discussed and tacit knowledge exchanged, intentionally generating moments where tacit knowledge is triggered through various influences by project members. This tacit knowledge can then be shared to the overall benefit of the project as well as the people involved to further their knowledge.

1.4 Summary and thesis structure

Tacit knowledge is an intangible good, however it is at the core of a knowledge based society. Throughout the following chapters tacit knowledge is assessed and put into context within a software development environment. A project is there to create something new and allows individuals to exchange their expertise with one another. Analysing the role of tacit knowledge is at the core of a successful project and enables a learning process within a group as well as an individual.

Following the introductory chapter, Chapter 2 will focus on the literature and conceptual framework for the assessment of tacit knowledge, software development and projects. It will also outline the conceptual framework, focusing on group and individual tacit knowledge. Chapter 3 demonstrates the methodology used to acquire and use the gathered data. Chapter 4 will focus on concepts, theories and results of the data evolution. The analysis and evaluation of the data is then shown in Chapter 5, followed by Chapter 6 where the results are put into context and a model will be built. Finally, Chapter 7 will conclude the thesis, where the key conclusions will be discussed.

Chapter 2: Theoretical Framework

2.1 Introduction

Knowledge is one of the most powerful tools in today's society. Alvin Toffler (1990) said we are now living in a "knowledge-based society, where knowledge is the source of highest-quality power". The power knowledge has over people as well as an organization is remarkable; it gives a competitive advantage which can be very difficult for other parties to catch up with. This advantage comes from the complexity of the product 'knowledge'. At its core, knowledge is the understanding of how something works. "It fundamentally involves the understanding of interrelations and behaviour. It is context-dependant" (Nonaka and Teece, 2001). The complexity of context, interrelations and behaviour can make the correct knowledge reception as well as creation a challenge. Incorrect knowledge creation can be a result of the complex process.

When endeavouring to show how tacit knowledge can be transformed for the benefit of a software development project, there are a variety of approaches. The deep mentoring approach proposed to by Leonard, Barton and Barton (2013) encourages observation, practice, partnering and joint problem solving (OPPTY). Using this approach in conjunction with a case study allows the observation of the project. This study will mainly use a case study approach with the focus on a primary research case study with the outcome of developing a conceptual model for analysis. The data gathered through a case study comes through documentation, archival records, interviews, direct observations, participant observation and physical attributes (Yin, 1994). The first section of the research will revolve around a careful perusal of the literature, with the aim of a conceptual model for analysis.

Throughout this section, an analysis of different literature will be presented with the goal of answering various questions in relation to tacit and explicit knowledge in software development projects. The first objective of the research is to investigate knowledge management. In more detail

knowledge assets and knowledge as human capital will be examined. Then, current research in tacit knowledge will be examined in detail. Theories of knowledge pioneers such as Polyene or Nonaka will be utilized to set a foundation, in response to the second aim, knowledge transfer and its effect on software development projects. Conceptualizing these theories of knowledge transfer and team tacit knowledge sets out to further understand the interaction of the tacit dimension. The natural decision making model will also be presented. The development of a conceptual model is the goal of the research and the literature sets out to help conceptualize the data and the creation of the model.

2.2 Knowledge Management

2.2.1 Knowledge Assets

When creating new knowledge one needs to first step back and assess the existing knowledge assets of the organization. “The basis of the knowledge creating process is knowledge assets” (Nonaka and Teece, 2001). It is essential for a project to know what the current state of knowledge, in order to explore and create new knowledge. Knowledge assets need to be managed, which can be done through a Chief Knowledge Officer (CKO). Knowledge assets are inputs, outputs and moderating factors of the knowledge-creating process. Hence, knowledge assets are at the core of an organization’s information transfer and therefore of a project, evaluation as well as usage (Syed-Ikhsan and Rowland, 2004). The impact of the asset on the organization is influenced by its quality.

Being one of the most important resources in today’s society, knowledge assets are a valuable part of an organization. Knowledge assets are organization specific resources that are indispensable to the creation of value for an organization. The custom knowledge assets of an organization partially represent the past, present and future value of a firm. However, “it should be noted that knowledge assets – especially routine knowledge assets – can hinder as well as foster knowledge creation. Organizations are

subject to inertia and it is difficult for them to diverge from the course set by their previous experiences” (Nonaka and Teece, 2001). Knowledge should be used in a forward thinking, effective manner. Knowledge assets are at their essence a form of knowledge possessed by an organization or in this case the project and its team members (Baldrige glossary, 2009). In more detail it is the information, ideas, learning, understanding, memory, insight and the cognitive as well as technical capabilities of a team. At times changing the knowledge transfer or creation channels help companies work more efficiently, but as stated above, changing the course of knowledge asset management can be a great challenge.

This challenge is created through a positive response from previous knowledge exploitation, where over time employees have trained to use a certain procedure which they have become used to. This inflexibility can lead to a slowing or even a ceasing of knowledge creation. “Successful experience leads to excessive exploitation of existing knowledge and, in turn hinders the exploitation of new knowledge” (Nonaka and Teece, 2001). The result of this problem can result in unsuccessful and non-usable knowledge assets. There are four types of knowledge asset creation, acquisition and exploitation.

1. Experimental

Experimental knowledge assets consist of shared tacit knowledge, which are built by means of shared hands-on experience among the members of the organization, and between the members of the organization and its customers, suppliers or affiliated firms.

Skills and know-how acquired through work experience by employees (Nonaka and Teece, 2001).

Prominent factors are: care, love, trust, physical knowledge (facial expressions, gestures), energetic knowledge (sense of existence, enthusiasm, and tension), rhythmic knowledge (improvisation, entertainment).

2. Conceptual

Knowledge assets which are conceptual are influenced by explicit Knowledge via images, symbols and language. Concepts are held by customers and members of organization. Nonaka and Teece state:

“As they have tangible forms, conceptual knowledge assets are easier to grasp than experimental knowledge assets, though it is still difficult to grasp what customers and organisational members perceive exactly.”

3. Systematic

Systemized and packages explicit knowledge are the main factors of systematic knowledge assets. These are explicitly stated technologies, product specifications, manuals and documented and packaged information about customers and suppliers. “This is the most ‘visible’ type of knowledge asset and current knowledge management focuses primarily on managing systematic knowledge assets, such as intellectual property rights.” (Nonaka and Teece, 2001). Therefore, patents, licenses and intellectual properties are also part of systematic knowledge assets.

4. Routine

Routine knowledge assets are “the tacit knowledge that is routinized and is embedded in the actions and practices of the organization” (Nonaka and Teece, 2001). Continuous exercise, certain patterns of thinking and action are reinforced and shared among organizational members. These include the shared stories of the organization as well as practical knowledge.

In their discussion of organizational knowledge, Brooking, Board and Jones (1998) argue that intellectual capital is really comprised of four things: market assets; intellectual property assets; infrastructure assets; and human-centred assets. The key insight to be extracted from this source is the idea that intellectual capital is really an asset that should be seen as any other asset: it has to be protected and value must be added constantly (Brooking et al., 1998). Intellectual capital is what allows each individual to be part of a project. It is what a person brings to the table and shares with project members, it is a person’s most valuable asset. Although the concept of

intellectual capital as an asset is very much integral to our earlier discussion of how it can be used to the full, mostly in the sense of how it must be cultivated or otherwise lost, the idea that intellectual capital can be reduced to an asset, and therefore as tangible as any other asset, collides somewhat with the notion that intellectual capital may also be a “space” or continuum wherein what matters is the process and not something tangible or discrete. In short, knowledge, the essence of intellectual capital, is not so much tangible things like datasets or mathematical equations as it is a particular disposition of the mind that makes learning its own reward. If this is done, then tacit knowledge is protected and becomes essential within a project.

2.2.2 Knowledge as Human Capital

If one views learning and knowledge as processes that never cease, then one essentially is creating an internal organization dynamic that never falls short of new trends or falls behind the learning capital of other organizations, such as human capital being equal. Create a workplace in which knowledge is never final, where all ideas are subject to proof, where all ideas and concepts are given consideration and respect, a spiral of knowledge can then be initiated that will take the entire organization to new heights by allowing weaker colleagues to access some of the implicit and tacit skills or proficiencies that more productive talents in the organization possess (Nonaka and Teece, 2001). It seems like a rather straightforward matter, but the concept of knowledge as space, the idea that knowledge perhaps is a sort of sensibility instead of a tangible set of facts, Figures, datasets and empirical conclusions, has not been universally accepted in the academic community.

For instance, Ulrich (1998) argues that organizational intellectual capital comes from employees’ competence and commitment. If both are present in sufficient measures, intellectual capital will automatically enhance growth. This sort of thinking, suffice it to say, suggests that an organization must give employees a salutary work environment that encourages commitment and that facilitates the growth in professional competence. It is an approach, however, that offers a rather simplistic template which, while

hinting at various things, such as the fact that an organization's strength derives from its ability to harness human capital, does not define how tacit knowledge can be turned into explicit knowledge. A more incisive way of looking at things is to ask how employee commitment to turning tacit knowledge into explicit knowledge can be achieved? Ulrich (1998) does not address this. In noting that, however, he indirectly underlines one simple fact, implicit, tacit knowledge and articulated, explicit knowledge are both constructed through the construction of competence. And that, inherently, means fostering an environment that makes learning accessible, fun, within the inherent limitations imposed by the task at hand, and comprehensive. Competence can literally be interpreted as another word for knowledge and knowledge is accrued, it cannot be magically transferred. Therefore, intellectual capital is really a manifestation of the learning capital, the ability to create a learning environment, of the organization.

Ulrich is not alone in thinking that learning is a process or a latent potentiality instead of a sensibility. Other scholars attempt to define and operationalize knowledge by suggesting that it is really about intellectual capital and knowing capability (Nahapiet and Ghoshal, 1998). In other words, some scholars seem to focus on the perceived capabilities or potentialities of a workplace collection, instead of focusing on creating a process that never ceases. They direct their attention to final outcomes, digging the latent intellectual harvest out of employees, however, rather than looking at creating processes which make learning an inescapable and never-ending feature of showing up for work.

Al-Ali (2003) submits that intellectual capital is really an omnibus of various things, it is employee knowledge, experience, brainpower, and it is the organization's databases, systems, processes and philosophy. This definition actually hints at something larger, processes and philosophies are the lifeblood of getting the most possible out of a workforce. Processes can unlock tacit knowledge and be the causeway linking it to explicit expression. And a philosophy of learning can make the transformation of the tacit to explicit a shared, collective endeavour. In that regard, intellectual capital is about more than just "smarts" or know how, it is about the processes, milieu

(contextual factors), and teaching routines that make intellectual capital acquire value and surmount obstacles.

The scholarly literature, increasingly, puts the emphasis upon recognizing that knowledge comes in two forms and requires two forms of communication. It is possible for a project to feature explicit knowledge transfer of a very detailed sort and yet the spaces that would be filled in by tacit knowledge are left unfilled. As a consequence, the project begins to crumble until such time as it falls apart utterly, all because particular subtleties that distinguish a successful project from an unsuccessful one are not incorporated into the knowledge transfer. As well, the success of delivery projects hinges greatly on the tacit knowledge gained from experience (Koskinen, 2000). As should be evident by now, tacit knowledge is something that emerges, fundamentally, from experiential processes or encounters. It all leads to a very significant finding.

Notably, a delivery project is dependent on people being taught the tacit knowledge they need to know. And that generally comes through experience or, at least, through actually seeing how something is done. And, not to be overlooked, it is imperative that as much tacit knowledge as possible be conveyed in addition to explicit knowledge. Some of these things cannot be written down; because of that, the only alternative is to utilize some of the rich media examples cited above (personal interaction or direct communication, or modelling). It can be laborious for people, and even time consuming, but it may be the only way of achieving ultimate success. Just as not all children and young people learn in the same way, so, too, may it be said that not one modality or medium can impart the knowledge that is vital to a software development project being completed properly.

Koskinen and Vanharanta (2002) write that tacit knowledge is the lifeblood of innovation at the earliest stages. When the product is actually being invented, when it is being conceived, that is when tacit knowledge, operating at a warp-speed, is achieving the greatest gains. The lack of formality, the absence of bureaucratic reifications, and the documentation piles that characterize getting a new project started at a large corporation are

generally much smaller at new or emergent businesses. With less bureaucracy and more informal lines of communication, the interaction between all members of the team can be defined by a generous give-and-take that may often seem indecipherable to those on the outside, but which acquires its own rhythm and beat to those who are working collectively on the project. It is now believed that a coaching style of leadership (be it be in development teams or in other contexts), coupled to a non-bureaucratic organization structure, can lead to greater utilization of tacit knowledge (Koskinen and Vanharanta, 2002).

Project management in the realm of software development projects – the sorts of projects which insist upon a seamless transfer of tacit knowledge storehouses to explicit knowledge – inescapably revolves around people management. A project manager or team leader must, if he or she wishes to create a dynamic learning environment, stress the need for all participants to learn from one another. There must be mutual support. There must be advice freely given and received. There must be constant questioning. And real issues and practical perplexities must be tackled via having people carry out their responsibilities in real conditions. In the properly managed development project, the team is a collection of learners who avidly explore ideas and encourage one another – in a positive fashion – to reassess and re-evaluate conceits, perspectives, and concepts they might otherwise not (Koskinen and Vanharanta, 2002). Project management is predicated on organizations creating an internal culture in which relational hierarchies are flattened, cross-cultural and interpersonal communication is facilitated, and managers see all members of the team as latent assets that need to be cultivated through a rigorous process of learning, interaction, and personal development. It's not easy to create such a culture – it requires a great deal of courage, in fact, for any project manager or for any organization seeking to produce the best (and most fearless) project managers – but it can be done. For that matter, it must be done.

Project management is one manifestation of strong organizational management, and organizations that thrive prioritize the human component or element in innovation and creativity. As a general rule, organizations set

the tone for research projects and, when the senior leadership is strong and intellectually curious, every software development project is invigorated and enhanced. To get the most out of people, and to create a truly vibrant learning milieu, it is vital that people work alongside others who share their concerns, want to see them succeed, and are motivated to get the most out of colleagues as well as themselves. One study carried out several years ago reports that people are five times more likely to turn to friends or colleagues for answers than to any other sources of information (Handy, 1994). Organizations need to cultivate project managers who value personal relationships and friendships. There must be a pervasive understanding on the part of everyone involved within the organization that supportive and collegial relations are essential if human resources in a group context are to be maximized.

Managing projects is intimately linked with larger organizational practices. Every project has its own timetable, its own schedule for deliverables, and its own required assets. But projects do not occur in a vacuum, they are the product of larger organizational trends, phenomena, practices and sensibilities. Friendship has to drive the organization, as it does any project. Although it falls outside the particular scope of this paper, one might argue that organizations, and even project managers, who want people to work alongside one another in a comfortable and friendly manner have to incentivize the idea of sharing with others and caring about others. People open up to, and turn to, those they trust in the workplace (Handy, 1994). Therefore, removing the systemic forces which might breed unhealthy competition amongst professionals has to be the key priority of anyone who cares about the wellbeing of the organization as a whole, or the wellbeing of any software development project. Although not stated directly in the literature, monetary incentives tied to serving as a mentor, or the tantalizing prospect of linking prior mentoring to professional advancement within the organization, might be one way of making people realize that projects are about the whole being greater than the sum of the parts.

Koskinen and Vanharanta (2002) long ago noted that organizational culture has to make room for individuals to be heard. Individual knowledge, if

an organization is to create an inventive and vigorous internal dynamic, has to become shared organizational knowledge through constant communication. Exchange, evaluation and integration must be the normative practices of any organization and these obviously have to extend to the level of the individual project team. Knowledge, in short, must be a social process where individuals are not kept in isolation (Koskinen and Vanharanta, 2002). How those overseeing any particular software development project choose to do this is presumably up to them, but team-building, partnering, and flattened hierarchies seem the logical places to start.

Tacit knowledge is a very proprietary form of knowledge. It is what distinguishes, and makes uniquely effective, the talented professional or the gifted artist. Due to it being such an intimate form of knowledge, it is the sort of knowledge that most people will find difficult to share, even if they can somehow muster the capacity to explain why they do certain things as they do. An organization in the highly competitive world of software development, has to find a way to get people to share their innate tacit knowledge. This is exceedingly difficult to do, in some respects, it is almost impossible to do. However, there are ways of shaping an internal culture, and the manner in which an organization carries out novel projects, that can greatly enhance the exchange of innovative tacit knowledge.

To get to the core of the matter, it is argued in some quarters that organizations need to create an innovative project team system wherein an objective is determined and everyone involved in the project is expected to share in the creative process. Everyone's ideas are included, everyone is involved in all steps of the process and everyone gets to ask questions and has every right to demand answers. The belief is that, by having all parties involved in the brainstorming and creative process, it becomes infinitely easier for tacit knowledge to be cultivated by one and by all. Additionally, because everyone is working together in a collective brainstorming context, implicit or tacit knowledge is more likely to be absorbed via assimilation. Innovation ideas are spread and latent talent is cultivated. Most of all, knowledge generation is democratized and the pitfalls that might otherwise arise from a few senior members monopolizing knowledge is curtailed (Zhi-

Guo and Cui-Jian, 2012). The preceding paragraph necessarily ties back to the earlier point about flattening hierarchies. An organization that wants people to be creative has to let everyone in on the process of creation. Not everyone will be able to contribute at the same level. Some will lack the innate talent; others will lack the drive and work ethic. But everyone should participate to the fullest of their personal capabilities. If people are allowed to involve themselves in idea generation, then even junior members or less talented team members can develop their own “knack” for doing things.

An elite organizational culture forever recognizes that learning cannot occur in a vacuum. People are constrained by their own intellectual limits, by their own experiential limits, what they have actually undergone professionally, and by their own ability to admit ignorance in various areas. A healthy organizational milieu will make positive reinforcement the cornerstone of project management. Project managers, taking their cue from their superiors, will emphasize friendship, conviviality, healthy competition, and shared proprietorship vis-à-vis the project and its conceits. Good quality social interactions and hands tacit knowledge transfer. More than that, however, the frequency of social interaction aids the acquisition and sharing of tacit knowledge because it creates better quality interactions and creates various opportunities for people to share ideas and to practice their craft in the comfortable presence of colleagues they trust (Ryan and Connor, 2013). Once more, managing any project involving people means creating a comfort zone that serves all of them well. Find the people who work most effectively with one another, build upon previous relationships or shared commonalities between individuals and make it plain that the general atmosphere will be one that fosters and facilitates the exchange of complementary knowledge and expertise. In effect, a major part of the challenge is ensuring that the right people are all together and working on the same enterprise. Interpersonal congruence, a concept noted more than once in these pages, is immensely vital to overarching success.

Finally, a few additional items must be sounded as this section draws to a close. Task interdependence appears to play a positive role in the extent to which team members communicate and to the extent to which tacit

knowledge is communicated or exchanged (Ryan and O'Connor, 2013). Project managers should, as much as feasible, create a network of interdependent tasks so that mentoring and modelling is facilitated. Individuals see how skilled individuals perform specific tasks with which they have little expertise or experience and internalize some vital tacit knowledge along the way. Overall, relaxed and confident social interaction appears to be the great means through which people are able to receive and transmit tacit knowledge (Ryan and O'Connor, 2013; Edmondson et al., 2003). As Dyer (1987) stated years ago, teams are collections of people who have to come together to achieve a common goal. Software development teams face challenges that far surpass those of most teams in any endeavour. Project managers, taking their cue from senior management, need to set themselves as examples of how to work generously and collaboratively with colleagues. If this internal culture is not established within groups and teams, then failure seems inevitable. Project managers, in that sense, are like coaches who recognize that internal social dynamics drive progress. Organizations want to consistently produce better results in the realm of research and development, have to understand that tacit knowledge and explicit knowledge will both not be optimized to the full unless teamwork becomes the default position of the group. As one can surmise, the same is true for the larger organization.

In general, there is a powerful symbiosis between successful group-level activities and successful organizational practices. Building a sense of fraternity and collegiality, creating a sense that the success of one colleague is a collective triumph, takes time and is deeply dependent upon successful orientation practices and hiring methods. Time and again, however, one thing remains steadfastly manifest: close relationships are vital to the sharing of tacit, complex knowledge (Granovetter, 1973). Close relationships appear to build trust and solidarity. They allow people to more unguardedly acknowledge their own limitations. And they allow for an unrestrained discourse because there is an implicit awareness on the part of both sides that neither side will attempt to discredit the other. Building relationships, in simplest terms, builds organizational tacit knowledge because it creates a

vital spider web of connections that turns everyone into an informational node. Project management should focus on what can be done to bring people together so that intellectual resources are utilised to enhance performance of individuals and groups.

As this section winds to its end, a few final points can bring into sharp relief the instruments which can facilitate outstanding software development project success. Leading the way is a sense that continuous social interaction facilitates people to share knowledge that might normally be difficult to articulate (Ryan and O'Connor, 2013). This conceit makes a great deal of sense, but other academicians submit that tacit knowledge acquisition actually rests even more on people seeking skilled performance in-person (Tsoukas, 2003). Project management, it would appear, rests on creating teachable moments wherein junior or relatively unskilled colleagues can watch highly skilled and experienced, or simply very talented, colleagues in action. What are they doing? How are they doing it? How do they organize and synthesize information? How do they make use of physical space when ordering or conceptualizing a problem? Seeing the right people in action should be mandatory for less proficient members of the team and it should be an absolutely integral part of how project managers organize time and group activities.

In general, one may contend that making the best use of tacit knowledge rests on creating intimate, interpersonal and multi-modal communication avenues that allow for subtle knowledge to be conveyed. Project management success may also be achieved to the full through what Clarke (2010) calls the Tacit Knowledge Spectrum Model. In essence, group discussions in professional settings are used to recall tacit knowledge and transfer this knowledge explicitly. It also calls for recalled tacit knowledge to be transferred experientially (Clarke, 2010). The conclusion of Ryan and O'Connor (2013) is that tacit knowledge acquisition is fundamentally a reciprocal process which may originate with individuals have peculiar storehouses of knowledge, but which gradually proceeds into becoming group and organizational knowledge through healthy social interaction. Bringing people together is what organizations should do, and it is what

project managers inescapably must do. Make all members feel supported and comfortable. Allow for mentoring and modelling. Keep communication informal and constructive. And, if the project managers are wise, allow everyone to have a proprietary sense of ownership in the process whereby their contribution will be counted as part of the whole.

From what has been discussed above, it is evident that project management success is contingent upon the capacity of the research team to exploit tacit knowledge. Since it is very difficult to articulate, multiple channels of communication must be embraced. Additionally, every effort must be undertaken to engender conviviality within the team so that people feel comfortable being unguarded or admitting to limitations. Project management in the realm of tacit knowledge acquisition is really about managing people and finding out what will get them to open up, what will get them to listen, and how different learning styles can be accommodated. At the end of the day, this really means understanding the people being brought into the group, both what they bring, and how they can be reached.

A conceptual framework for a study of how clearly one is able to define and grasp the notion of a “learning space” that is dynamic and fluid from an individual and group perspective. The introduction of the Nonaka and Konno (1998) study offers a conceptual paradigm that will frame this chapter as it explores efficacious ways of transforming tacit knowledge into explicit knowledge. The conceptual framework here is built up from particular components of the theory of ‘Ba’ internalization, externalization, socialization, and combination, in addition to Ryan’s TTKM as well as Clarke’s Tacit Knowledge Spectrum, which can be directly correlated to the case study. It is a provisional framework for analysis, and these components derive largely from concepts and models in the current literature. Even if they do not do so in a direct manner, they may well indirectly manifest themselves in the literature when other scholars discuss such things as learning environments, knowledge exchange and how they are nurtured through iterative processes and collaborative efforts. Ryan’s transactive memory and how knowledge is exchanged within a group will aid in manifesting knowledge exchange in software development projects. Finally, Clarke’s ‘Tacit Knowledge Spectrum’

provides the framework of the conceptual model for analysis, enhanced by previous findings. To summarize, this chapter will assess the literature concerning tacit knowledge from an individual and group perspective.

2.3 Tacit Knowledge

Defining tacit Knowledge is not easy as some might suppose due to the sort of instinctual knowledge which is intangible, it is therefore rather difficult to come up with a truly satisfying definition for it. Moreover, as the next several pages will show, it has many nuances and various stands, nonetheless, a workable definition is possible and will further be demonstrated.

Notably, Ryan and O'Connor (2009) insist that tacit knowledge is the aggregation of "articulable tacit", individual and goal-driven "expert" knowledge that exists at the team level in constituent parts embodied by the different members of the team. In other words, it is the sum of the knowledge that all parties in a professional research and development team possess (please see Ryan and O'Connor, 2009). This seems like a useful definition, but there are problems it poses.

As a pioneer in the field, Polyani (1958) describes tacit knowledge as being the knowledge we possess but cannot express in an explicit or articulable way. Tacit knowledge is precisely tacit because it is hard for us to describe or define: it is largely intangible. Therefore, it is not clear what Ryan and O'Connor (2009) are really trying to say when they describe "articulable tacit" knowledge. Tacit knowledge, it may be contended, is knowledge that involves modelling or showing someone something, but it is frequently so subtle and so much a matter of "touch" or "feel" that it is virtually impossible for even a skilled, seasoned professional to convey it to someone else. Thus, to focus our attention on the brief definition above, tacit knowledge is, indeed, part of the aggregation that is "knowledge," but it is its own discrete part. Therefore, while a starting point, the definition offered by Ryan and

O'Connor (2009) is unsatisfying and misses the mark, when utilising Polyani's (1958) definition.

There are other definitions and overviews that shed more light on the concept in question. Tacit knowledge is, at least a few might argue, what the human body and mind absorb unthinkingly and unreflectively during the course of a day. It is the sum of relationships or invariances that the environment displays. It is, it would seem, what reality enforces upon us and which we, arguably unconsciously, perceive (Reber, 1996). With this in mind, learning is something that involves more than merely reviewing a page or a draft or more bluntly a checklist. And knowledge is fluid, multi-varied and multi-dimensional: it may be both tacit and explicit, both instinctive and carefully learned. When seeking excellence in an organization, the proper utilization of tacit knowledge is perhaps the one ingredient that distinguishes companies which are able to execute sophisticated IT projects, and those that fall short.

Explicit knowledge generally entails written instructions; guidelines; protocols; step-by-step processes. It is information or knowledge that can be conveyed through written or verbal instructions and the other party has a reasonable chance of apprehending what is being transmitted. Tacit knowledge, on the other hand, is more difficult – more nebulous. It operates at the innate or instinctive level. It is knowledge that is hard to convey, often harder than hard to understand. It rests, literally, at the level of the sub-conscious and involves mental schemata that most of us take for granted – or have never bothered to deconstruct. It is, above all else, knowledge that we cannot think about when we are doing the activity that demands it; it occurs instinctually and manifests itself with great rapidity. Seeing knowledge as an asset within an organisation, differentiating between tangible and intangible assets, explicit and tacit, helps explore the two dimensions. The constitution of knowledge and its being such a vast storehouse of know-how, understanding how to convey or articulate tacit knowledge may be the most significant thing any organization can do in order to achieve a true learning culture. Nonaka and Takeuchi (2001) provide a scheme in which tangible and intangible assets are demonstrated. These should aid in further the

understanding differences between explicit and tacit knowledge and how these assets are managed.

The value of knowledge can be immense, and can, as stated previously, give an organization a competitive advantage. To understand the difference between tangible and intangible assets is a crucial part in understanding and protecting. "An important difference between intangible and tangible assets is the availability and enforcement of property rights." (Teece, 2001) Unlike tangible products which can easily be physically separated, such as real estate by walls or fences, intangible assets are difficult to separate in a tangible way.

When exploring the subject of tacit knowledge, there is the temptation to view it as one part of a seamless continuum that is, bluntly stated, merely all of the various stages of knowledge: elementary knowledge, knowledge you can explicate and articulate, and finally knowledge which features a practical understanding of how something works. It is commonplace for scholars to view tacit knowledge as the informal part of a learning continuum that eventually proceeds to formal and practical learning (Wang, 2009). Thus, this sort of thinking, without saying so directly, seems to hold that tacit knowledge is just explicit knowledge in a primitive state. Once our learning has struggled and examined its way to true insight, we will thus be able to articulate it like we can all other forms of knowledge. However, to take this view is to perhaps reduce an enormously complex matter to a dangerously simplistic view.

Comparing and contrasting scholarship notes, there are some skills, especially physical skillsets, which do not lend themselves to codification. Beyond that, there are elemental mental schemata at work which appear to be a sort of subsidiary awareness which allows us to interpret the world in a manner permitting us to operate in it. We groom this knowledge through action and experience and repetition (Busch, 2008). What scholarship in this vein is evidently making plain to us is that not all knowledge can be turned into explicit knowledge. However, in a broader sense, it can still be turned into comprehensible knowledge, which is, for all intents and purposes,

essentially explicit, through a learning culture that urges people to experience different activities, processes and roles to learn from.

To fully appreciate the expansive and immensely complex cognitive process which yield up tacit and explicit knowledge, one must recognize that tacit knowledge, and all knowledge in a broad sense, is a subset of human intellectual capital and therefore an asset. The value and meaning of understanding what intellectual capital is, and how it can be harnessed, is part of understanding how tacit knowledge can be harnessed and transformed into something even greater. Optimizing intellectual capital in the workplace environment really is only achievable if non-verbal opportunities for information and knowledge transfer are made possible. Because of its inexpressible or intangible nature, tacit knowledge fairly demands that people engage in modelling and constant practices to properly explore tacit knowledge.

By this point, one can see that tacit knowledge is knowledge that is almost in-articulable. It is knowledge that instinctively allows people to see patterns and rhythms that cannot be apprehended by mechanical means. It is a knowledge base that people cannot express in words and is an aggregation of all experiences we have undergone in life. This largely distinguishes it from explicit knowledge, which, arguably, is more predicated on formal learning. Tacit knowledge is really the interstitial space between the formal rules we are all supposed to follow; it is a series of conventions that may vary according to the circumstance. For example, when people are urged not to walk too close to others on the street, there are no specific rules which indicate where you are to be positioned relative to others during the course of your travels. However, there is an implicit understanding of what the concept means – and an implicit recognition of what “don’t walk too close to others” means in the context of ordinary social life (Collins, 2010).

Tacit knowledge is fundamentally driven by experience. Because of this, it is hard to embody in formal language and often is conveyed using metaphors, drawings and other informal means of conveyance. It is practical know-how and, because it is, it may be the most challenging sort of

knowledge to transfer from one party to another (Koskinen and Vanharanta, 2002). For many years, scholars have emphasized the idea that the absorptive capacity of a professional – which is to say, the ability of a professional to assimilate and use new bits of information – is rooted in earlier knowledge (Cohen and Levinthal, 1990). If one has experience in engineering or design projects of a relatively similar nature, for instance, then it becomes easier to tackle questions or issues that appear to offer confounding and novel subtleties. Experience is the essence of tacit knowledge, and a smart organization will do everything in its power create opportunities for new employees to tackle software development challenges in which they can assimilate and learn new knowledge without fear of reprisal if initial efforts fall short.

Tacit knowledge is almost an amorphous concept in many ways. However, it does appear to have discrete components or elements. Maybe the most significant, at least in the sense that it appears to most directly address the matter of problem-solving, is the technical element that manifests itself in tacit knowledge. Tacit knowledge exists in techniques and skills, craftsmanship, and the course of production will reveal this technical or practical element. Tacit knowledge is about “knack”, skill, or about making adjustments or changes in the course of a project. Technical expertise, when it falls inside the realm of craftsmanship, relies heavily on vision and what might be called “feel” or intuition. Thus, there are elements that cannot be coded or written down adequately in language. Furthermore, anything that entails a “knack” or a personal vision of an evolving project is going to be highly individualized and unique to the person in question (Zhi-Guo, and Cui-Jian, 2012). There is probably no universal means of passing along an individual “knack” to others, it is why the sum of tacit knowledge is so hard to articulate and illustrate, but practice can allow each person to develop their own personal “knack” or “feel” so that daunting tasks become somewhat less daunting in the end.

2.3.1 Elements of Tacit Knowledge

Huawei et al. (2002) offer an exhaustive and richly detailed overview of what they perceive tacit knowledge to be. Their conceptualization is comprised of five elements: the technical, cogitative, experimental, emotional, and faith element. These suggest the astonishing broadness of tacit knowledge: anyone who wants to tease optimal knowledge and performance out of project team members needs to take a comprehensive approach to employee development and empowerment.

Looking at each of the five elements above in turn, the technical element is best described as being the craftsmanship or hands-on “knack” for bringing projects to a neat completion that is possessed by the most accomplished professionals (Huawei et al., 2002). This subject area will be described in more detail later in the review, it will therefore not require an extensive description at this point. Nonetheless, the technical element is perhaps best understood as the “craft” of software development or any other type of enterprise project. People with talent and ability take a project and shape or mould its technical features so that the vision is brought into being. There is a certain deftness of touch in this area that either takes great talent, or years of experience to develop. People enraptured by the “big picture” may downgrade the craftsmanship that turns a vision into reality, but the subtle skills of the craftsmen, or craftswomen are the essence of tacit knowledge. Therefore, a smart organization will always stress the practical necessity of learning by trial and by error, which is another idea which will be explored in greater depth later in this paper.

Besides the technical element of tacit knowledge, there is the cognition element. This is a rather mysterious aspect of tacit knowledge. It entails internal knowledge structures, the ability to identify and solve problems, the capacity to absorb novel forms and bits of information, and a complex symbiosis between memory, thinking and learning. Group-level cognition involves the informational context of the enterprise, how information is exchanged so that it can be processed by all parties, and how information or new knowledge is accepted (Huawei et al., 2002). The

cognition process in the human mind is a remarkably incomprehensible thing. It is hard to encapsulate or describe. Nonetheless, tacit knowledge arises from cognition that is rooted in absorbing, synthesizing and ordering new knowledge. By creating an internal environment that privileges these things, and that makes knowledge acquisition as democratic as possible, even less-talented associates can gain greater competency than they would otherwise have imagined possible.

The third element of tacit knowledge, as described by Huawei et al. (2002) is the experience element. This maybe the most straightforward part or element of tacit knowledge, it is the accumulation of mental, physical, cognitive processing, and sensory experience, pretty much every sensation or trial or encounter that has led to the present level of knowledge an individual or a team possesses. It may be argued that experience is something that should be shared amongst all members or parties in any project enterprise because that allows for the leveraging of all available resources while saving time (for more on the experience element, please see Huawei et al., 2002).

The fourth element of tacit knowledge is the emotion element. Here, it can be presupposed that interpersonal congruence comes into play in the sense that employees' emotional makeup and status, their hostilities, preferences, passions, feelings of collegiality, fuel knowledge acquisition commitment and mental perspectives. It is the most flexible of all the five elements, and can be controlled and channelled through energetic leadership (Huawei et al., 2002). Although often overlooked when examining tacit knowledge, the emotional element is the one element that can open up the democratic pathways that allow for tacit knowledge to be spread and leveraged within a software project team or within an organization. It should never be discounted.

Fifthly, there is the matter of the faith element. Faith is the belief of the employees on the basic knowledge of the value of the enterprise. It shapes commitment levels and the willingness to persevere in the face of obstacles and hurdles. It is the shared values that guide the team, the project, and the

overarching organization that oversees the team and breathes life into it. It shapes the tolerance for innovation and different views and the embrace of innovative thinking (Huawei et al., 2002). Tacit knowledge in a lot of ways, to the extent it cannot be articulated or even much understood, is very much the embodiment of an act of faith: we cannot explain why we do certain things, except that we do them in spite of ourselves. An organization or a team that wants to get the most it can out of tacit knowledge needs to kindle faith in order for ultimate success to be reached.

Overall, tacit knowledge is not easily synthesizable and is not easily transferable. It is the sort of knowledge that must be seen and felt in order to be internalized; it cannot be wholly reduced to a checklist. Because it is inarticulable in many respects, tacit knowledge has to be teased out via various multi-modal devices. It requires a social component that relies heavily on informality and symbiotic relationships. Overall, tacit knowledge is the “knack” some people possess, it is the elemental component of knowledge that comes with experience. For that reason, an organization or research team has to create an internal dynamic that allows people to feel comfortable dispensing knowledge relating to their accumulated experience. And that is not always an easy thing to do.

2.3.2 Tacit Knowledge in an Organization

Clearly, the transformation of tacit knowledge into explicit knowledge in software development projects is critical: an organization must be able to create excellence and expertise within its project teams, and this does not happen if a vital bridge is not erected between the two types of knowledge. It may be said that not everyone will be an expert, or naturally gifted or adept at something, and this is where the tacit knowledge of elite performers needs to be translated for those who do not have such a bedrock. A cursory glance at the subject suggests that effective social networks and mentoring and modelling practices are critical to achieving success. With the proper practices, and bit by bit, that which is difficult to apprehend, or difficult to

capture in words, can be understood by those who have a greater capacity for learning and accomplishment than they might envision. It seems that software development projects clearly have to feature close feedback, intimate ties between colleagues, and a recognition that different people will learn optimally via the use of different modalities. Additionally, there does need to be space and time for experimentation and practice. Although it may seem esoteric, people who may believe otherwise can build up their own tacit knowledge foundations and then use those foundations to gradually foster a growing literature of explicit knowledge within the organization.

McAfee (2003) writes that checklists alone cannot lead to successful IT projects and to fruitful implementation. Leadership is a subtle craft and a subtle art, and proper knowledge transfer does not easily lend itself to simply interpreting checklists. This is especially so when there are subtle gradations with regards to when a project is truly done right – or may simply be done “mostly” right. So many pitfalls – inertia, misspecification, resistance, misuse and non-use – can spell doom for any IT initiative. Small mistakes in preparation, planning, leadership style, and even timing can lead to serious mishaps. In the end, even the most detailed and prescriptive projects can quickly run aground (McAfee, 2003). This is the essence of why tacit knowledge is so important, and why turning tacit knowledge into explicit knowledge, to the fullest extent possible, can be so beneficial: there are decisions which have to be made, adaptations which must occur, that will involve a bedrock of knowledge that is not likely to be found in any checklist. If that knowledge which is difficult to articulate can be turned into something articulable, some of the loopholes cited above can be closed. Moreover, even if it is not possible to close every loophole, the more an organization is able to make comprehensible tacit knowledge, the more it will be able to create orientation programs and training programs that teach project workers and all employees to develop the right habits of mind, the right competencies, and the necessary “read and react” skills to meet evolving circumstances properly. But it all starts with tacit knowledge and how tacit knowledge is utilized and understood.

Tacit knowledge in an organization is as much about collective or organizational memory, institutional memory, as it is about anything else. Experience is multi-faceted and allows individuals and organizations to see the correlations and dependencies between entities and components. The existing knowledge structure, its richness and breadth, makes it possible for connections to be drawn and for causal linkages to be illuminated (Lyles and Schwenk, 1992). Thus, tacit knowledge in the context of an organization is made evermore potent through individuals and the organization consciously striving to accumulate a storehouse of practical knowledge that is predicated on individuals being granted opportunities to bolster their practical exposure to software projects, components, movable parts, and mathematical theorems. The more people in an organization are immersed in the practical workings of a problem or set of issues, the more likely it is that the organization will possess superb tacit knowledge stores.

Tacit knowledge in an organization would appear, given what has been described in the preceding pages, as something that very much encompasses experiential knowledge. There are some things that cannot be conned via reading about them, one must undergo them. Indeed, one might argue that the ultimate distinction between talented amateurs who think they know what they are doing, and talented professionals, is that the latter have practical experience in formulating the answers and outputs for software development projects of the most challenging sort. Scholarship reveals that the quality of experiential tacit knowledge learned from so-called “enterprise activities” ultimately determines the quality of experiential knowledge that lies at the core of tacit understanding (Zhi-Guo and Cui-Jian, 2012). This has undeniable implications for any organization that wants to create a culture of learning and professional growth. Most notably, it would appear that junior employees need to be exposed to a variety or medley of different experiences in which they will be called upon to make executive decisions and creative selections. A graduated process is in place that demands individuals gradually overcome increasingly complex and difficult tasks. If properly done, in a manner that is conscientious and incremental, the quantity and quality of experiential learning can surely be impressive. And,

suffice it to say, any organization can stand to bolster its existing stores of tacit knowledge.

An organization's tacit knowledge is really the one thing that allows it to survive in head-to-head competition with other entities. According to some of the literature, tacit knowledge evidently accounts for roughly 90% of the total knowledge of an organization (Zhi-Guo and Cui-Jian, 2012). Explicit knowledge is but a fraction of the actual knowledge that powers organizational success. As one celebrated CEO has argued, tacit knowledge is the true intellectual capital of an organization and it provides nutrition to the roots of the trees (Zhi-Guo and Cui-Jian, 2012; Jing, 2003). Tacit knowledge in any organization will have to be embedded in communicative practices, in social activities, in group endeavours, and in streamlined feedback loops that bring people together rather than pulling them apart. It can be nourished by standardized organization programs or plans, but the interstitial spaces really will have to be filled in by the creation of conventions and cultures that facilitate constant mentoring, constant monitoring, constructive feedback, and people working with people. An organization which fails to do these things really cannot find success in the face of other organizations more committed to optimizing in-articulable insights.

Clarke (2010) also states that constant reflection must be a part of best practices for any organization desiring of bolstering tacit knowledge internally. In effect, something happens, a problem arises, and a cycle of reflection and consultation occurs.

For tacit knowledge stores to be accessed and utilized optimally, the organization cannot simply create an overarching framework of best practices that encourages learning: it must locate knowledge-hungry and exemplary types in close proximity to younger, more impressionable types whose attitudes and behaviours can be easily influenced. This is plainly ideal for software development projects, but the only way to create a group-level or meso level culture that facilitates knowledge transfer is by having the skill to identify the right people for mentoring and leadership roles – and then finding a way to match people congruently so that the internal learning

dynamics of an group (like the organization in a larger sense) are boosted to their fullest. To the extent that hiring, promotion and incentive systems can identify the right people for demanding research and development projects where knowledge transfer and tacit knowledge expression are integral, it can be argued that human resources professionals actually play a huge role in the innovation and research branches of any organization.

2.3.3 Knowledge Transfer in an Organization

Having a head start is usually connected to knowing or having something other companies do not; in business this is the competitive advantage, which can allow an organization to stay or become profitable. Today knowledge and the ability to create and utilise knowledge is seen to be the most important source of an organisation's sustainable competitive advantage according to management scholars (Nonaka and Teece, 2001). In order to achieve a competitive advantage companies therefore need to create new knowledge, which can be in a tangible or intangible form. Companies need knowledge in order to exist, since at its core an enterprise is a knowledge centre, where ideally it is created, exchanged and utilized. The reason for their existence is the constant production of knowledge. Not generating, obtaining and managing knowledge correctly risks profitability and sustainability of the firm and according to Teece and Nonaka goes against its purpose.

To understand the nature of a firm a reflection and analysis of the environment is needed. Adapting to the environment and solving problems, in order to achieve a specific goal, organisations are seen as information-processing machines (Nonaka and Teece, 2001). Aiming at serving to solve a problem in its environment an organization's knowledge should be a direct mirror to its surroundings and adapt to it accordingly. Exchanging knowledge with its environment gives a firm the opportunity to position itself according to market needs. Correct and constant knowledge creation is essential for the organization's survival.

Durability of a firm is directly connected to knowledge creation, which is a constant exchange between the organization and its environment. As continuous process, organizational knowledge creation is never-ending and never stops to upgrade itself, this interactive spiral process takes place in both intra- and inter-organisationally. The knowledge transfer goes beyond the organisational boundaries and inter-plays with other organisations, with the goal to create new knowledge (Nonaka and Teece, 2001). Noting the knowledge exchange between different companies, the blurred lines of intangible assets and its property rights are challenged. The obvious interpretation of organizations needing each other in order to create meaningful knowledge, questions the flow, ownership, management as well as original creation of the knowledge asset.

As previously stated knowledge is dynamic and is question to interpretation by its environment, therefore knowledge is changed depending on the 'new' context it is interpreted in. "By means of this dynamic interaction, knowledge created by the organization can trigger the mobilization of knowledge held by outside constituents, such as consumers, afflicted companies, universities or distributors." (Nonaka and Teece, 2001). The individuality of knowledge gives a large range of opportunities to receivers via an interpretation of core information, obtained by previous created knowledge, adapted to a concept of new context and belief. Newly created knowledge can be seen as a knowledge asset, which sets the basis for future knowledge creation. The main sources needed to create organizational knowledge assets are human resources, which need to be recruited and managed correctly to achieve its goals. "Knowledge creation is an individual activity and that the primary role of firms is to apply existing knowledge." (Nonaka and Teece, 2001). New staff adds to the knowledge resources, which can then be shared and exchanged with the organization. The fusion between the different knowledge sources of each individual of an organization gives a basis for new knowledge creation.

The interaction between individuals within an organization also greatly depends upon the management of the assets at hand. Knowledge assets, not being able to be bought or sold, must be constructed and used inside of

the organisation for the full value to be realised (Nonaka and Teece, 2001). It is critical to support the knowledge creation and distribution process within the organization, to unfold the full potential of the new asset. Ensuring an asset is fully utilized by an organization does not depend on it being internally or externally used. "In organizational knowledge creation, neither micro nor macro dominates. Rather, they interact with each other to evolve into a bigger self." (Nonaka and Teece, 2001). When creating knowledge the interpretation and facility to use it in different fields and ways is crucial. Freedom of creation gives opportunities to become a larger, more sophisticated knowledge asset.

The environment and human resources playing a vital role in knowledge, organizations have the opportunity to build and sustain a firm specific knowledge base. Building their own knowledge assets from experience, an organisations' tacitness is what makes experimental knowledge assets firm-specific (Nonaka and Teece, 2001). These help organisations gain unique, and difficult to imitate, resources, with the goal of producing a sustainable competitive advantage. The organization specific knowledge needs to be nurtured in order for it not to create information, instead of justified true belief. A organization's knowledge base is adjustable depending on their environment. The free nature of knowledge allows expansion in all directions. "As knowledge has no boundaries, any form of new knowledge can be created regardless of existing business structure of the organization" (Nonaka and Teece, 2001). Adding new knowledge to a firms existing base as well as adapting it to their current structure, permits companies to expand their reach to new organisational as well as business territories. The new found territories support future knowledge creation and as previously stated can provide the firm with a competitive advantage.

At its core knowledge creation is a human activity, where the quality is reliant on the people who construct it as well as the environment it is built in. "To create knowledge, organizations should foster their members' commitment by formulating an organizational intention, as commitment underlies the human knowledge-creating activity" (Nonaka and Teece, 2001). Nurturing the knowledge generators within an organization through support

and committing them to the organization's goals is crucial for the quality of the generated knowledge. Being part of organizational goals and seeking for ways to reach those gives members the opportunity to find new frontiers within their knowledge base.

2.3.4 A Dynamic Environment to Create and Share Tacit Knowledge

Using 'Ba' as a platform for knowledge exchange, it serves as a basis for knowledge creation, which is a continuous task. Nonaka and Teece (2001) developed the SECI model to further understand the way knowledge moves across and is created by organizations. The model represents a spiral of knowledge creation, allowing to be repeated infinitively, enabling knowledge to be expanded horizontally as well as vertically across an organisation (Nonaka and Teece, 2001). However, for the spiral to work in many cases, actors need to share space and time, 'Ba', in order to efficiently create a spiral of knowledge exchange. Supporting the idea of a meeting being 'Ba', the knowledge spiral, especially socialization and externalization, is the most efficient when teams come together. To further examine and understand the model, first an explanation of the four paradigms is needed.

One definition of knowledge is that it is a shared space wherein relationships emerge (Nonaka and Teece, 2009). It is not really tangible, but spiralling knowledge which is constantly changing and building result in new levels of tacit knowledge. It is a self-transcending and ever-spiralling evolution. Knowledge is internalized via socialization and experience. It is most commonly tacit knowledge. Then knowledge is externalized and combines with other externalized forms of knowledge, interaction with others and the environment, and the spiralling process proceeds anew (Nonaka and Konno 1998). Knowledge is thus not a set of facts and Figures, it is not a set of statistics or applied conceits, but a "space", which Nonaka and Konno identify as 'Ba', a mental flexibility and ongoing dynamic process, that allows for new insight to be constantly generated. If one embraces the concept of 'Ba', as expressed by Nonaka and Konno (1998), then one essentially is

arguing for a learning culture in which processes are constantly iterative, marked by close communication, marked by modelling, by mentoring, and by incessant experiential inputs that lead to outputs which bolster and build knowledge. Knowledge is really a process and not a terminal result. Nonaka and Teece differentiate between four different elements of 'Ba'.

Originating 'Ba' are the individual and face-to-face interactions, which create the basis of originating 'Ba'. Experience, emotions, feelings, and mental models are shared, hence the full range of physical senses and psycho-emotional reactions are grasped. These include care, love, trust, and commitment which are at the centre of human coming occasion. This allows tacit knowledge to be shared, in that context of socialization.

Dialoguing 'Ba' our collective and face-to-face interactions, which enable mental models and skills to be communicated to others. This supports the theory of externalization, producing articulated concepts, which can then be used by the receiver to self-reflect. A mix of specific knowledge and capability to manage the received knowledge is essential to consciously construct, rather than to originate new knowledge.

Collective and virtual interactions are in focus when discussing systemising 'Ba'. Offering a visual, written, context for the combination of existing explicit knowledge, online networks, groupware, documentation or databanks, knowledge can easily be transmitted to a large number of people.

Finally, exercising 'Ba' allows individual and virtual interaction, which is often communicated through virtual media, written manuals or simulation programs. Nonaka and Teece (2001) state, "exercising 'Ba' synthesized the transcendence and reflection that come in action, while dialoguing 'Ba' achieves this via thought. "

'Ba' often acts as an autonomous, self-sufficient unite that can be connected with other 'Ba' to expand knowledge, it seems to work in a similar way as a modular system or organization, in which independently designed modules are assembled and integrated to work as a whole system (Nonaka and Teece, 2001). Seeing 'Ba' in a work, software development context, and a translation to meetings can be made, where knowledge is exchanged,

enhanced and built upon. All four stages of can be found in a meeting, from originating, to dialoguing and systematic to exercising 'Ba'.

2.3.5 Channels for the Exchange and Sharing of Tacit Knowledge

There are several channels which allow tacit knowledge to form into explicit knowledge. Nonaka and Teece (2001) show the SECI as a way to analyse tacit knowledge with an environment of 'Ba'. These allow a categorization of different ways in which tacit knowledge to be shown. To begin, the process of socialization is the conversion of tacit to tacit through shared experiences. Found in apprenticeships, learning by doing is at the core rather than a theoretical approach. Self-transcendence is vital for socialization to work, since it can only be shared through direct experiences. Interaction with customers is therefore at the forefront for companies to take advantage of tacit knowledge transfer. Following, externalization is the process of converting tacit to explicit knowledge, where a person articulates knowledge and shares it with others in order to create a basis of new knowledge for a group. A successful conversion of tacit to explicit knowledge greatly depends on cohesive metaphors, analogies or models to be understood by a group of people committed to understand and internalize knowledge. The explicit to explicit conversion is called combination, where explicit knowledge is set into more complex explicit knowledge. Gathering explicit knowledge from in- or outside of the organization, editing or processing allows new, more complex, knowledge to be created, which can then be transferred to co-workers. Finally, the process of internalization is integrating explicit knowledge to make it one's own tacit knowledge. It is the counter part of socialization but from the apprentice point of view. This internal knowledge base in a person can set off a new spiral of knowledge, where tacit knowledge can be converted to tacit, explicit and combined with more complex knowledge.

Using the four paradigms of the SECI model demonstrated in Figure 1 below, a spiral of knowledge creation is found where internal tacit knowledge is the basis for socialization, externalization and combination to create more

complex knowledge. When trying to convert tacit into tacit or explicit knowledge, social interaction is crucial for a successful and well understood knowledge transfer. Furthermore, accessing the spiral in a meeting context, continuous interaction facilitates the spiral of knowledge to commence, build up and re-launch the process. This concept is the foundation of the knowledge exchange within the analysed meetings. For further comprehension Nonaka's spiral of knowledge diagram is represented below.

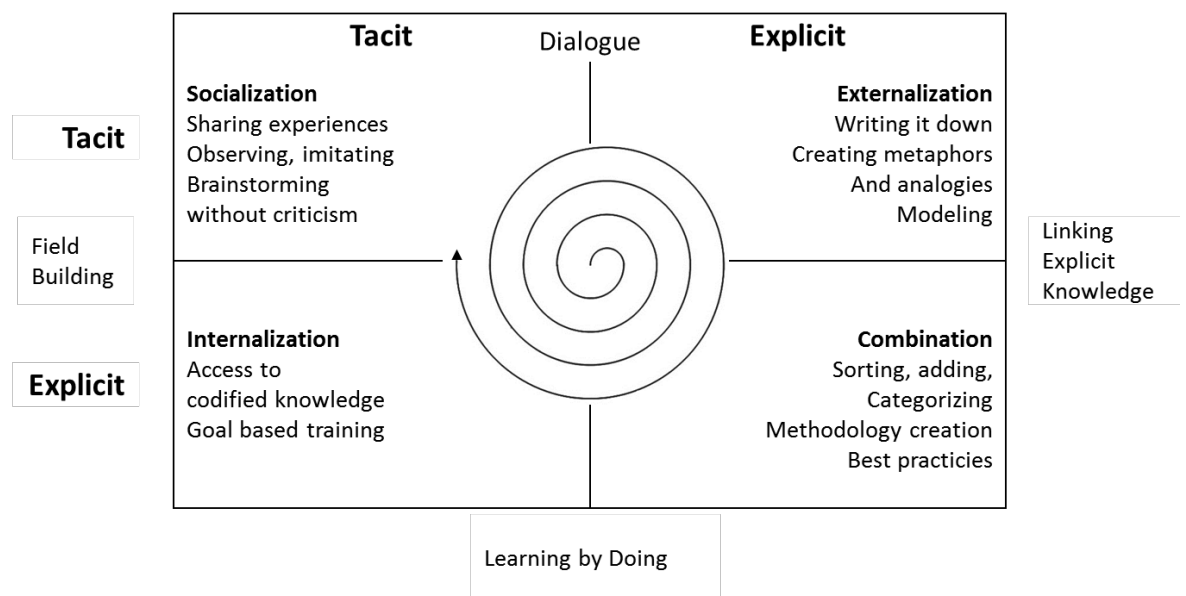


Figure 1 - Spiral of Knowledge (Nonaka and Teece, 2001)

This spiral including the element of 'Ba' allows the basic assessment of tacit knowledge. Understanding how tacit knowledge surfaces and spirals over time in a project enables an assessment of the conversations held in meetings, and when which form of transfer is taking place. This then allows Clarke to further investigate the transfer of tacit knowledge.

2.4 Analysing Tacit Knowledge from an Individual Perspective

One of the most central questions of the research is whether it is possible to recognize and harness tacit and explicit knowledge within a conceptual model in software development projects. There are several

different approaches where scholars have attempted to grasp the process of converting tacit to explicit knowledge. However, to begin one needs to look back at the early writings of knowledge management research and assess main concepts such as Polyani (1966), who lined out that “we can know more than we can tell”. This takes knowledge to a highly esoteric level, where knowledge is the whole; and internal as well as external knowledge can be seen as the parts. Nonaka and Takeuchi (1995) took the concept further and created the knowledge creation spiral and separated knowledge as the whole into two separate parts, the explicit and tacit. Standing in contrast to each other, Nonaka’s view of explicit and tacit standing alone and Polyani’s argument of them being two dimensions of one, is a key dichotomy in the tacit literature (McAdam, 2007). However, later scholars such as Clarke (2010) argue that a successful conversion of tacit into explicit knowledge is possible and therefore lays the foundation of his ‘Tacit Knowledge Spectrum’ on the ‘Spiral of Knowledge Creation’ seen in Figure 2 below.

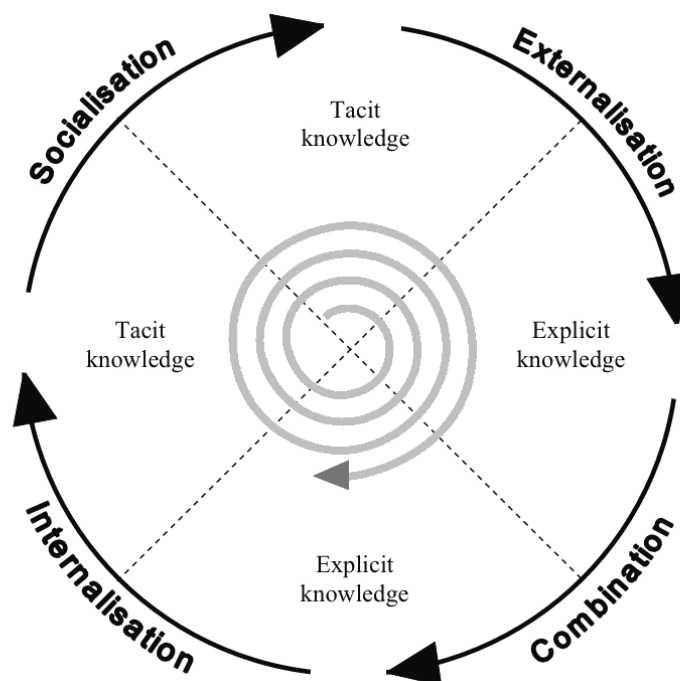


Figure 2 - Spiral of Knowledge Creation (Nonaka and Takeuchi, 1995)

Furthermore, when looking at tacit and explicit knowledge interaction, tacit and explicit knowledge react to one another, however on opposite sides of the spectrum. When tacit knowledge is converted and used it becomes explicit knowledge, at this impasse tacit and explicit knowledge overlap. This can be seen as the learning and conversion process, where personal knowledge is processed and identified through personal reflectiveness, then brought to the surface in order to be changed and edited to make it explicit and hence comprehensible to the receiver. Tacit knowledge being in the semiconscious and unconscious part of one's body and ending in a structured, codified and accessible part. This process takes place in a spectrum, where knowledge exists and flourishes (Leonard and Sensiper, 1998).

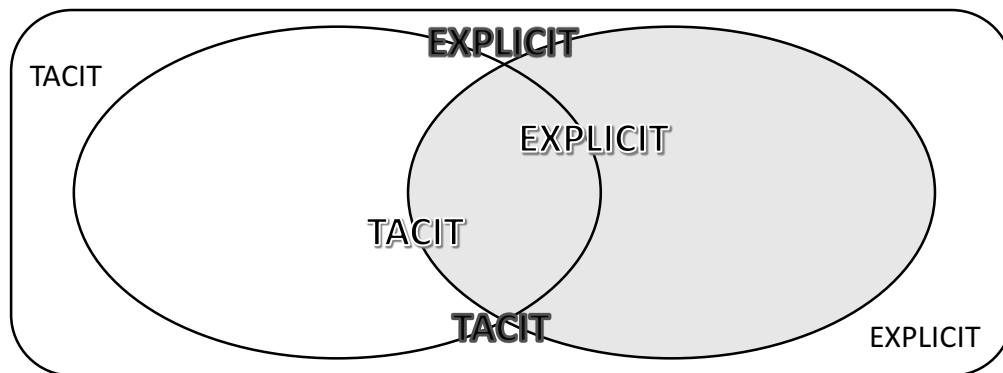


Figure 3 - Explicit and Tacit Knowledge as Two Dimensions of Knowledge (McAdam et al., 2007)

Tacit and explicit knowledge live in a spectrum where they can interact and grow, which can be demonstrated as the macro perspective, which aids the big picture view of the process. The aim of Clarke's model, the tacit knowledge spectrum, is to target specific elements of tacit knowledge, in order to facilitate tacit knowledge transfer for organizations in a timely and resourceful manner. As we have previously explored, Huawei (2002) gave us several elements of what he perceives tacit knowledge to entail. A different micro perspective of tacit knowledge is demonstrated by McAdam (2007), who proposes the idea of epitomes of tacit knowledge (table 1). He identifies seven different epitomes, intuition, skills, insight, know-how, beliefs, mental models and practical intelligence, which are, although tacit knowledge is seen as a personal process, often referred to as

a more collective form of tacit knowing. These make up another part of the tacit knowledge spectrum, which are shown and referenced by various scholars in the proposed model of McAdam, shown above in Figure 3 and further elaborated by Clarke.

Table 1 – Epitomes (Clarke, 2010)

Intuition:	Intuition is expressed as directly knowing our learning without conscious reasoning or making choices without formal analysis (Brockmann and Anthony, 1998).
Skills:	Skills can include negotiation, physical, coordination or cognitive skills. This is perhaps the epitome that is most used without any form of definition.
Insight:	Insight is used as an understanding, often in sudden form but also as “glimpses” into knowledge (one’s own or others).
Know-How:	Know-how is often expressed as the ability to put know-what into work which is to great extent the product of experience (Brown and Duguid, 1998).
Beliefs:	Believes used as a set of understandings that reflect our perspective of the world.
Mental Models:	Mental models are cognitive structure is formed by the abstractions of experience. They reflect our perspectives of the world around us (Giunipero et al., 1999).
Practical Intelligence:	Practical intelligence is expressed as “a person’s ability to apply components of intelligence to everyday life” (Somech and Bogler, 1999).

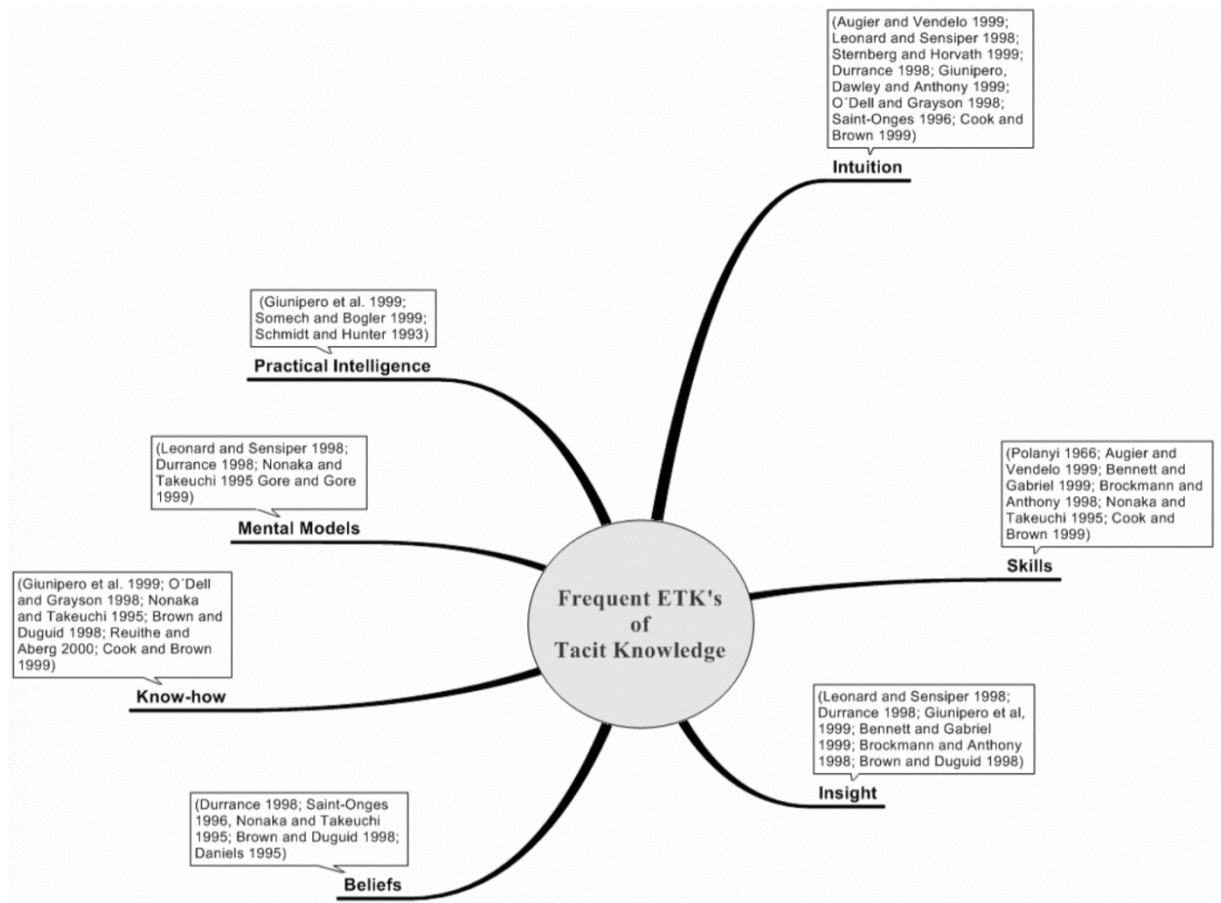


Figure 4 - Epitomes of Tacit Knowledge (McAdam et al., 2007).

Throughout Clarke's case study research, tacit knowledge was always present, however some of it could not be turned into explicit or only transferred through experimental teaching. As previously stated he found elements of tacit knowledge, confirming McAdam's et al. (2007) approach towards epitomes (Figure 4). Identifying tacit knowledge which could be easily recalled enter into explicit knowledge, tacit knowledge which can be transferred into tacit knowledge, tacit knowledge which needs a trigger for recall and finally tacit knowledge that is hidden to the individual, are at the core of his findings. Labelling these as 'the tacit to explicit element', 'the tacit to tacit element', 'the triggered response element' and finally as the 'unknown tacit element', it is crucial to comprehend these elements, to further construct of the tacit knowledge spectrum (table 2).

Table 2 - Tacit Elements (Clarke, 2010).

The Tacit to Explicit Element	An individual is able to recall information such as a process or working practice about a defined subject.
The Tacit to Tacit Element	An individual is not able to process knowledge into explicit form.
The Triggered Response Element	An individual is only able to recall knowledge through a triggered response.
The Unknown Tacit Element	An individual's knowledge base, which is not easily extracted, but needs intensive triggers for recall.

The process of reflection is a central issue when talking about tacit knowledge. Meetings, being a learning place, are a fruitful platform for reflection and interaction between individuals in order to trigger elements of tacit knowledge. When assessing McAdam's dimensions of tacit knowledge, one could argue that the spectrum is the place of reflection. This process can be seen as a loop of knowledge being inserted, which is turned into tacit knowledge and through the processes of reflection turned into knowledge output. This process is demonstrated in Figure 5 below.

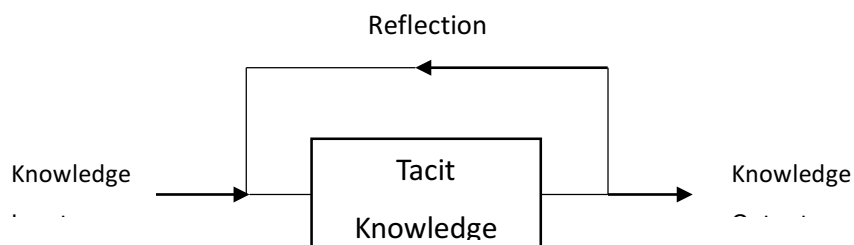


Figure 5 - The Reflective Process (Clarke, 2010)

The triggered response element identified by Clarke, shown in Figure 6, represents the process of the recollecting tacit knowledge, which is able to surface within certain environments such as group discussions or pressure situations.

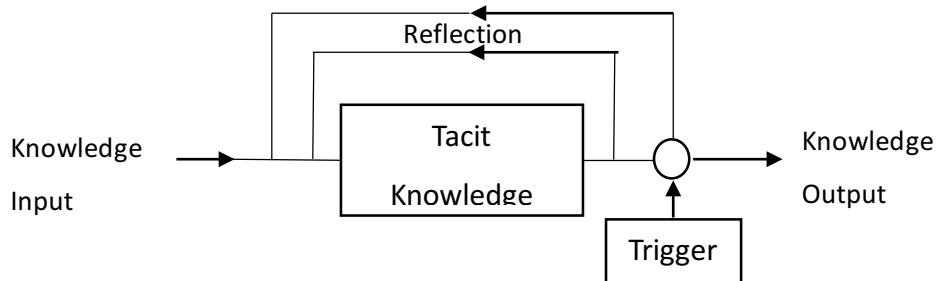


Figure 6 - Triggered Response Loop (Clarke, 2010)

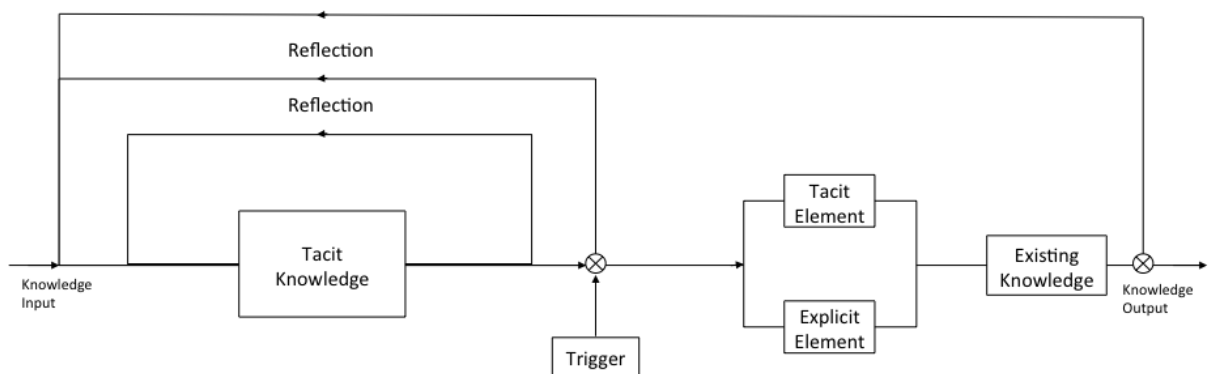


Figure 7 - The Tacit Knowledge Transfer Model (Clarke, 2010)

The final step before identifying the tacit knowledge spectrum as a whole, is the tacit knowledge transfer model (see Figure 7). The tacit and explicit elements are integrated into existing knowledge and another reflective cycle is added to the process. With each cycle a deeper, more profound layer of tacit knowledge is revealed.

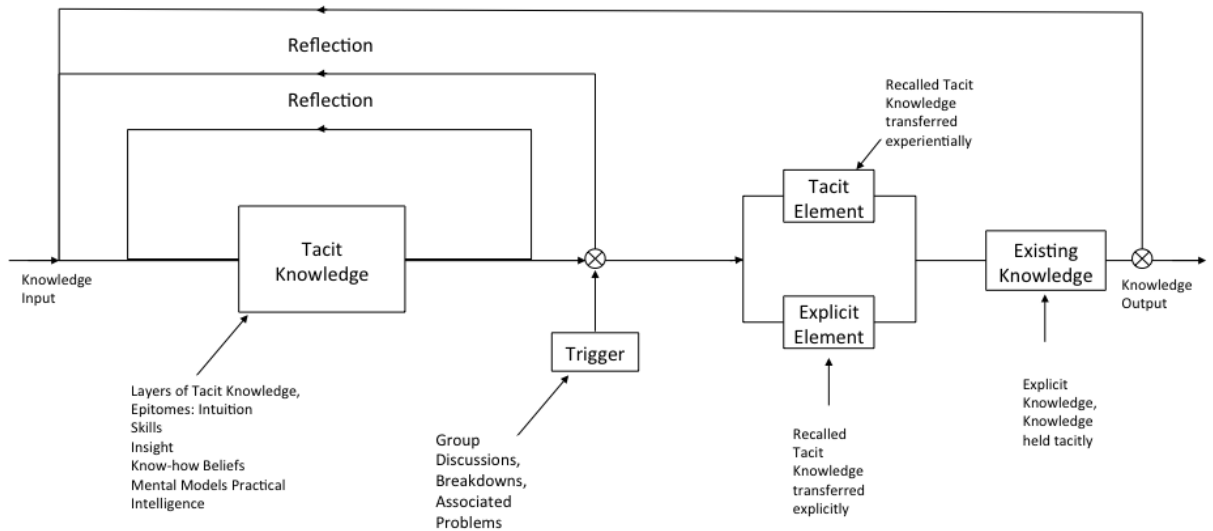


Figure 8 - The Tacit Knowledge Spectrum (Clarke, 2010)

The tacit knowledge spectrum, represented in Figure 8 above, exhibits the different layers and processes in the tacit knowledge cycle. Elements such as tacit to tacit, tacit to explicit or triggered response help understand and reveal the different layers of tacit knowledge exposure. Through Nonaka's SECI and Ba it enables a more detailed view of knowledge transfer. The concept of knowledge triggers, where knowledge surfaces through an external influence is introduced.

2.5 Team Tacit Knowledge

Creating a model for team tacit knowledge and how it is measured is a great challenge in the knowledge field. Scholars such as Sternberg, Busch or Ryan and O'Conner have tried to define the different types of team tacit knowledge and have created methods to extract it. Sternberg et al. (2000) have created a psychological approach, referred to as the 'Yale group approach' by Busch, defining tacit knowledge as a practical intelligence feature. It is acquired through a minimum of environmental support and aids in peruse of personal goals. Most importantly, it is not about what an organizational member knows but their ability to utilize knowledge, 'know-how'. Taking into account that tacit and explicit knowledge are constructed

through the interplay of personal and social interactions (Kelly, 1955/1991), it is essential to compare and contrast group tacit knowledge and individual tacit knowledge.

In projects, teams are assembled to create a well-functioning group of experts in order to complete the task at hand. Mohammed and Dumville (2001) define a team mental model as 'an organized understanding of relevant knowledge that is shared by team members'. Each team member having a specific skill set, with their tacit knowledge differentiating from one another, allows a more complete exchange of tacit knowledge. Ryan and O'Connor (2008) define team tacit knowledge as the aggregation of articulable tacit, individual, goal driven expert knowledge to the team-level where different member of the team possess different aspects of tacit knowledge. This assumption sets the basis for their TTKM (Team Tacit Knowledge Measurement) model, and confirms Nonaka and Teece's (2010) idea of the SECI, where knowledge is created through learning from others.

Team tacit knowledge gradually prospers over time by the interplay of knowledge and group cohesion. The more people work together, the more they understand an individual's skill set and how it can best be utilized within the group. Berman et al. conducted a study on the NBA (National Basketball Association), assessing played minutes in a season with a player's experience, revealing that a player's success is related to the increase of team tacit knowledge, hence experience and the cohesion within the group is directly related to success. Berman et al., as well as the Yale group have founded their studies on proxies' attempts to address and challenge the unobservable character of tacit knowledge. Ryan and O'Conner have used the theory of proxies in order to create the TTKM.

The TTKM seeks to reveal the further understanding of team tacit knowledge in software development projects. As previously stated Ryan and O'Conner base their work on the Yale group's proxy approach, which commences with the differentiation of experts and novices. Experts, unlike novices, possess task performance expertise relative to their domain (Ryan and O'Conner, 2008). Building from this assumption three main assumptions

need to be taken into account in order to build the TTKM in a software development environment. First, team tacit knowledge reflects domain specific practical knowledge, which differentiates experts from novices. Secondly, the TTKM needs to measure the tacit knowledge of the entire team, taking the weight of the member into account. Finally, only tacit knowledge at the articulate level of abstraction can be taken into account.

Having set a basis for analysis, Ryan and O'Conner chose to focus on Kelly's (1955/1991) Repertory Grid, rather than the Yale group's situational judgement, in order to reveal personal knowledge and enter private worlds. The grid is classified in two ways, first to illuminate elements, which are a person's observation of the world and the classification of these elements. The links a person constructs between the elements and their classification also plays a vital role in the comprehension of tacit knowledge within groups. In addition, using these classifications to compare and contrast expert and novice knowledge can show different degrees of tacit knowledge as well as separate the levels of domain specific practical knowledge.

To get started, transactive memory systems (TMSs) were conceived several years ago by Wegner (1987). Ryan and O'Connor (2013) summarize Wegner's work by noting that members of "long-tenured" groups tend to rely heavily on one another in order to obtain, process and communicate information from various distinct knowledge domains. Wegner (1987) enforced the idea that knowledge specialization is actually greater in such groups that feature strong transactive memory systems. In such a group, there is immediate expertise recognition and group members will consult with other group members when they have concerns about acquiring relevant information. They will also, as expected, evaluate that information on the basis of the source involved (Wegner, 1987; Moreland, 1999). In a software development team that is functioning in a healthy manner, the trust amongst members will be implicit. That is to say, each party within the team will believe in the competency and veracity of the person aside from him or her – or anyone he or she wishes to consult about peculiar questions. The memory of the group is all about people relying upon people and knowing that their respective inputs are valued and appreciated. Once more, this sort of

collective memory draws heavily upon social interaction and upon the use of social interaction to tease as much tacit knowledge out of all parties as possible.

Having delineated the broad contours of Wegner (1987) and Moreland (1999) into their own work, Ryan and O'Connor (2013) present their own synthesis of what an effective transactive memory system should look like in software development projects. Principally, they argue that an effective team will coordinate the differentiated or specialized knowledge that defines each of the group members. Knowing who has the knowledge, and then coordinating this knowledge, is the essence of maximizing or optimizing group learning or knowledge, particularly in software development (Ryan and O'Connor, 2013). When in the centre of a difficult and demanding project cycle, it may be put forth that great leadership entails identifying individual competencies and adjusting roles and responsibilities in light of this. A well-functioning transactive memory system builds up implicit trust through interpersonal congruence, through getting to know each member and what he or she is comfortable doing and actually capable of doing, creating a flattened decisional hierarchy in which those with capabilities in various areas are allowed to step forward and seize the initiative on matters that refer to their area of specialty. A controversial view to optimise tacit knowledge and its transfer is rooted in knowing people.

Organizations can create cultures and routinize best practices. They can even arrange project teams or pods in a manner that is satisfying and effective. But, at the end of the day, an organization can only do so much: those who are actually involved in the software development project at the ground level are the ones who are going to have to facilitate and nurture effective transactive memory and tacit knowledge acquisition. Transactive memory, wherein people rely upon each other in an interdependent manner, is a group-level process, having previously defined it as the 'meso-level' process, whereby software development project team members work informally via interpersonal communication. Work teams that interact regularly tend to perform at a much more productive level than dyads which do not interact constantly (Liang et al., 1995; Moreland and Myaskovsky,

2000). The key is to make time for interpersonal communication and to foster a sense of togetherness when embarking on a project. Absent this sense of togetherness, success can be almost impossible to achieve because so much tacit knowledge is transferred informally and in collegial settings.

Transactive memory systems are absolutely one way of achieving strong collective memory and expertise in a project that demands the aggregation of many different skills and specialties. However, more than a particular system, a software development project must rely on people working together as one. It is stated in the literature that group member familiarity, communication volume and frequency, and “task characteristics of interdependence, cooperative goal interdependence and support for innovation” were vital to TMS and, by extension, elevated productivity (Ryan and O’Connor, 2013; Lewis and Herndon, 2011). If at all possible, ideally a software development project team should be drawn from a professional group that has complementary skills and pre-existing professional and personal relationships that heighten comfort and faith. Admittedly, that is not always easy to achieve, since finding individuals who possess both exceptional technical skills and an easy familiarity with one another is a dyad that most organizations struggle to find. The amount of scientific knowledge accumulated in an individual is at best perceived only intuitively by his more experienced peers.

The meetings most people attend generally have characteristics belonging to more than one of these three prototypes according to Ryan (2013) (conference, school, workshops). These can be seen in Figure 9.

- 1) Team tacit knowledge has been (and is being) created by team members
- 2) Individuals draw from the team tacit knowledge and create their own tacit knowledge. This is a background process which is dynamic and reciprocal relying on constructivist situated learning
- 3) This knowledge is re-integrated and becomes individual knowledge
- 4 & 5) As individuals interact, informally and face-to-face, tacit knowledge is acquired and shared and a TMS is also developed. TMSs allow for

knowledge to be stored and shared, and are therefore both dynamic and static.

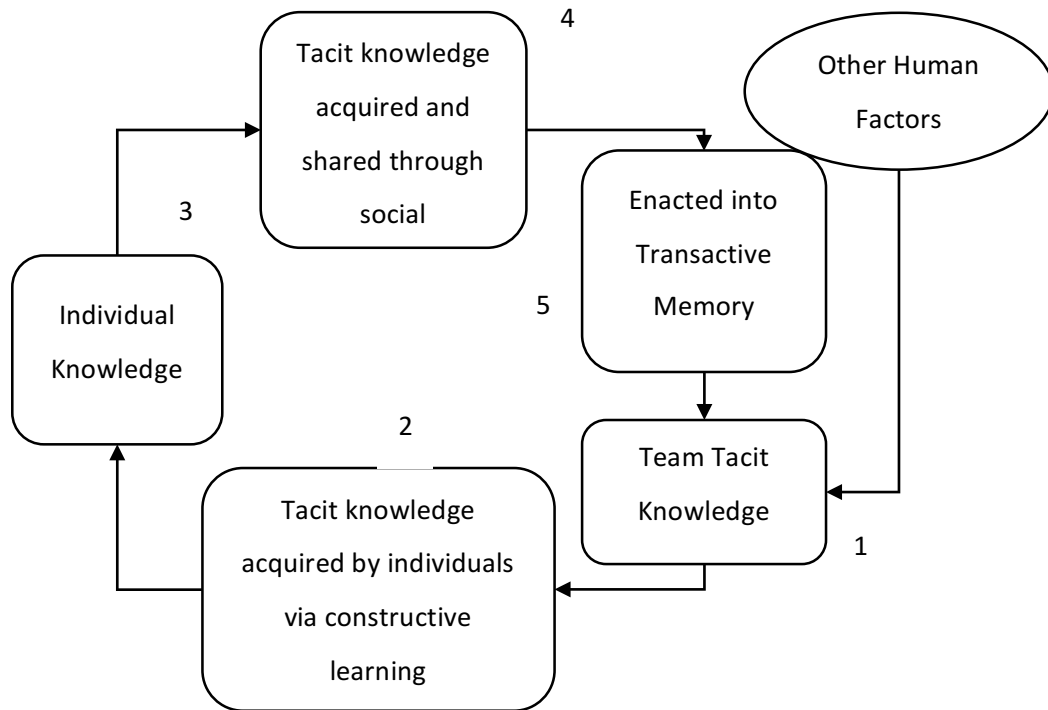


Figure 9 - Theoretical Model for the Acquisition and Sharing of Tacit Knowledge in Teams (Ryan, 2013).

Nonetheless, transactive memory systems that cultivate interdependence and complementarity appear to be a snug fit for many software development projects. Research in recent years by Akgun et al. (2005) stresses that a TMS paradigm has a greater impact on team learning, on speed-to-market, and on new product success when the task complexity was of a greater magnitude. Tacit knowledge may not be easily explicable, but having teammates leaning on one another does appear to allow for sufficient knowledge transfer and clarification to expedite success in challenging group tasks. As Ryan and O'Connor (2013) note, software development teams work on very complex tasks that feature many interacting elements that demand coordination and integration. A TMS framework could be one way of making what appears incomprehensible a bit more comprehensible. There is certainly nothing to indicate that it will make matters worse.

2.6 Decision Making and Tacit Knowledge

The exchange of tacit knowledge during a project is a continuous process which aids in creating new individual and group tacit knowledge. This new gained knowledge can be used on an individual or group level to enhance the project. Decisions made during a project, should be informed by utilizing knowledge gained from other project members. As previously discussed there are several ways to exchange tacit knowledge within a project, however the way decisions are made through this gained tacit knowledge is another aspect of tacit knowledge acquisition and its output. Naturalistic Decision Making (NDM) hinges on understanding experiences gained in real-life and how they are utilized rather than using a specific decision making set (Klein, 1999). Naturalistic Decision Making is often used in specific situations, such as dealing with uncertainties, using context and learning, non-defined goals or insufficient or incorrect information, limited time and changing conditions, according to several scholars these are major issues in IT projects (Boehm, 1991; Heemstra and Kusters, 1996; Schmidt et al. 2001). NDM can often be used subconsciously by experienced decision makers in such situations as well.

The NDM model, seen in Figure 10, hinges on a simple yes and no process which commences with an individual point of view of the situation within a changing context. This implies that a change needs to be made within the current process presented. The individual consequently evaluates whether the situation is familiar, at which point tacit - internal- knowledge comes into play. The situation is recognized and evaluated by the individual as well as by the group. At this juncture, the question of whether the expectancies are validated or not can relaunch the cycle. Either the situation needs to be reassessed, where more information needs to be found or an action to solve the issue is created as well as its implementation. Once again, if the action discussed will not work it needs to be modified or the situation reassessed. When an action is created and it does not need to be modified, it can be implemented in order to solve the issue at hand.

Using the NDM model within a tacit knowledge decision making context allows a systematic review of a situation within a meeting. In combination with tacit knowledge theories within software projects, from individuals as well as groups, a step by step evaluation of problem solving can be used. Exploring a conversation during a meeting and applying the NDM will give insight on tacit knowledge from an individual and how it helped solve an issue, people involved in the situation as well as how the decision was made to resolve the issue.

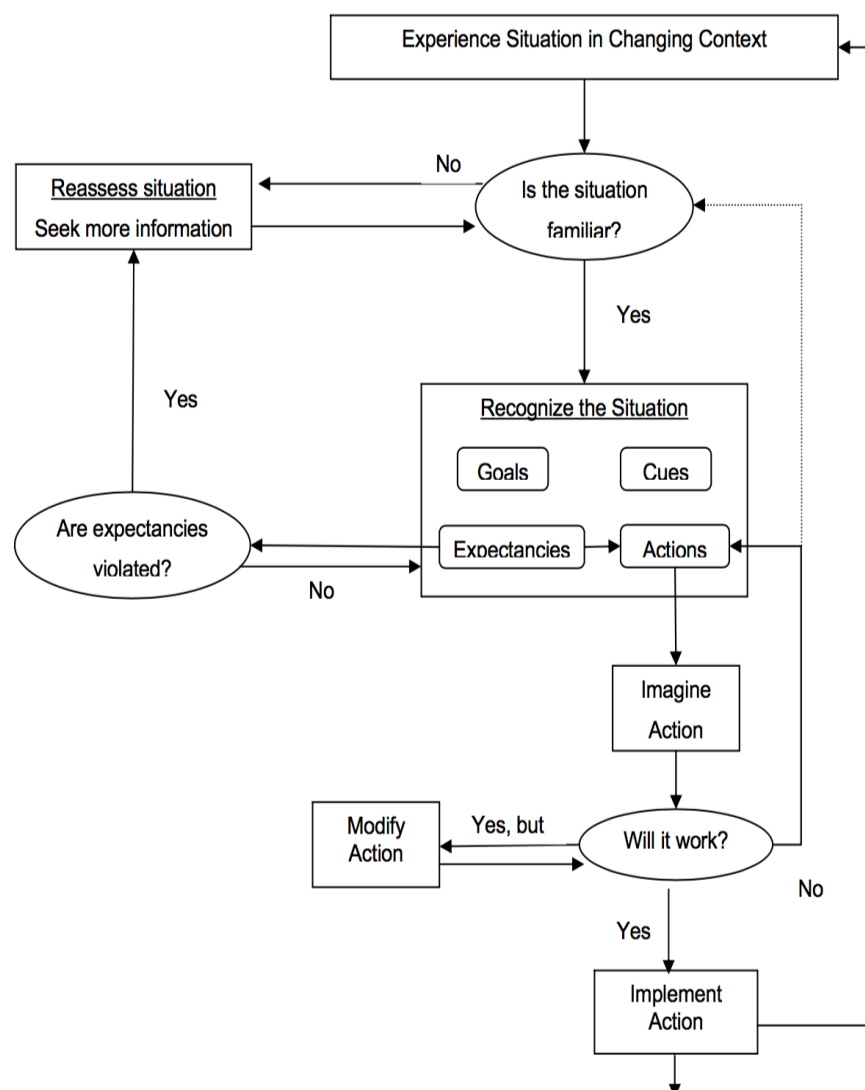


Figure 10 - Natural Decision Making Model (Klein, 1999)

2.7 Conclusions

When looking back on the preceding pages, it is clear that tacit knowledge needs a dynamic environment to flourish. It is the imagination at work. And all imaginations, no matter how creative or fertile, need the comfort of support, constructive feedback, and professional guidance. By allowing ideas to be visually presented, or even presented in some sort of short-hand, the steps towards explicit knowledge become shorter. It becomes easier, in effect, for people to take an inchoate idea and give it order and rationality. It is not an easy process. It takes time, hard work, intellectual application, and a great deal of patience. It may proceed by small increments, or it may explode in a series of large ones. Conversely, it can just as easily go in reverse. But if the internal culture is adequate, then great things are certainly possible.

One may also surmise that discussions and exchanges with subjects in the case study will shed even more light on what further extensions are needed to really illuminate best practices for making the inarticulate fully articulate. There are, though, broader steps which can be taken which will cover a great deal more ground.

Specifically, research into how to convert tacit knowledge into explicit knowledge should focus upon strategies which have been adopted in the modern-day classroom. Maybe one of the reasons why tacit knowledge is so difficult to transfer into explicit knowledge is because everyone learns in his or her own unique way. Therefore, when exploring the challenges or obstacles posed by software projects, an individualized and customized approach may be in order. What makes some people learn faster than others? Will a multi-modal approach suffice? Will a collaborative approach work? What components should be in an orientation training program for young people or older people entering the organization? And what should the organization do as far as teaching people to teach others? What are the interpersonal competencies that are needed so that a project head can bring everyone onboard and get the most out of one and all? These are areas of discussion and exploration that should be investigated to the fullest.

Progressing research in this area is not expected to be easy because the topic area is highly esoteric and intangible. After all, the mind is what is being discussed and researched, and the human mind is still one part of the physical human anatomy about which even the brightest medical minds know relatively little. It is well and good to say that a learning environment is the way to go. It is even better to argue that a customizable, individualized learning environment is the way to go. However, how should all the pieces fit together? Are there certain modalities as far as information presentation that might facilitate transforming tacit knowledge into explicit knowledge? The more one pores over the literature, the more it is plain that no one seems to have a truly comprehensive or prescriptive template for what the perfect learning environment will be. That is possibly not a surprise: tacit knowledge is somewhat of an area of mystery, and turning what we cannot even articulate, in many instances, into reproducible and effective explicit knowledge, which is the sort of knowledge we can hopefully memorize and internalize, is a daunting task. Nonetheless, developing a model to help software development project but communicate tacit knowledge, is a fundamental step in ensuring that companies can meet the fluid and twisting demands of extraordinarily complex and multivariate engineering projects.

One further item to be explored is the subject of managing intellectual capital and organizational knowledge in such a fashion that proprietary interests are not lost. A organization that wants to encourage learning and constant growth must also be wary of the danger posed by failing to protect its intellectual assets from harvesting from other firms. Communication internally must be free-flowing and constant; communication to the external world must be constrained and carefully guarded. With this in mind, any effort at facilitating a learning environment has to be married to a concomitant desire on the part of the organization to do everything in its power to make sure that knowledge transfer only occurs between internal contracting parties. One area of exploration that captures my personal attention is looking at the possibility of a meeting being seen as a learning environment and whether other project managers see similar problems within the meetings.

Similarly, literature into the subject area can reveal the sorts of managerial competencies that are inestimably valuable when creating a learning culture. Is there something about the organization's management team that might enhance learning? Are they transformational leaders? And what is the role of transformational leadership in creating a learning culture where tacit knowledge becomes explicit knowledge and back again? Workers are commonly confronted with their own deficiencies when identifying why a software development project falls short of expectations. Without diving too deeply into the available literature, one has the inescapable sense that transformational leadership, persistent modelling, shortened feedback loops, and the elimination of bureaucratic layers are all staples of good management in this area. Thus, good management will include people who are receptive to the idea that a business place is also a classroom that treats employees as students. Senior managers, therefore, must be teachers in their own right and must pass on what teaching skills they can to subordinates.

Progressing the research, on a practical level, will not stop with capturing only the areas of inquiry illustrated above. There is also the need to cultivate relationships with the case study organization so that people are as forthcoming as possible. There is no magical elixir for achieving this, but constant communication and attention to needs and concerns can absolutely ensure that the organization will be receptive and supportive – and also open and ingenuous when discussing the subject matter. It is hoped that a closer affinity with the subjects of the study will shed invaluable light into confounding variables that might frustrate efforts aimed at bolstering learning within the organization.

Important though it is, even a sweeping literature review cannot, given its secondary nature, achieve truly profound insights: at some point, the scholar must wade into the fray and craft original or primary research. The literature review above has, as much as anything else, provided ample terrain for a researcher to explore the applicability of a bevy of theories to actual, every day, organizational life. This is where a case study should be,

and will be, appended to the above-mentioned literature review and which is aimed to produce a conceptual model for analysis.

To explicate or illuminate the connections between the theories presented above and lived circumstances in the “real” world, a case study of an existing organization is well in order. Since Nonaka and Konno (1998) are Asian scholars at least partially invested in scholarship involving non-UK businesses, there needs to be some consideration granted to whether or not their theory, and its theoretical corollaries, can fully be of utility in the context of a UK firm aspiring leadership in the software development sector. The literature review presented in the pages above makes it abundantly clear that learning is complex, fluid, ceaseless, and seems inextricably bound to human relations and to the internal culture of an organization. Thus, cultivating relationships within the particular firm in question, including some that are already present, will serve as a vital entrée into how this organization appears to honour the dictates of Nonaka and Konno (1998) and to what extent it might have still more to learn from their concept of ‘Ba’. Shaped by various questions arising from the literature review, including a few enumerated in this final section, the case study will comprise unstructured interviews with subjects with the goal of the conceptual model for analysis. How were project teams organized? Were they organized on the basis of interpersonal congruency with an eye towards partnerships and mentoring? What efforts were undertaken to achieve the transformation of tacit knowledge into explicit knowledge? How do the questions above translate into Clarke’s ‘Tacit Knowledge Spectrum’? The literature review can, and already has, provided us with some answers, the unstructured interviews and their translation are aimed to provide us with the rest.

The literature review provides a macro-level overview of how tacit knowledge can be turned into explicit knowledge. Any subsequent case study must, by extension, focus on learning more about the dynamics within the software development project. It must explore the longitudinal evolution of the project and what challenges emerged along the way. Any inquiry must be tailored towards getting a sense of how those involved ultimately perceived the project’s attempts to facilitate knowledge transfer and idea

generation. From the literature, we learn that project success is made possible through the creation of a certain organizational sensibility, rather than narrowly focusing on final ends or results (Nathalie and Ghoshal, 1998). Therefore, utilization of an unstructured interviews where questions arise will establish how a software project team's members perceive continuous learning and whether they are devoted to outcomes or to the process of learning and communicating itself and its transfer to the model.

A conceptual framework for a study of this sort hinges upon how clearly one is able to define and grasp the notion of a "learning space" that is dynamic and fluid. Path-breaking research by Nonaka and Konno (1998) offers a conceptual paradigm that will frame this entire paper as it explores efficacious ways of transforming tacit knowledge into explicit knowledge. This paradigm is used by Clarke (2010) in order to construct his 'Tacit Knowledge Spectrum'. The theory can become diffuse and amorphous due to its vastness. As stated previously, it is imperative to look at particular components of the theory of 'Ba', internalization, externalization, socialization, and combination, and integrate them to the case study. It is anticipated that all four of these components will manifest themselves, from time to time, in the prevailing literature, as well. Even if they do not do so in a direct manner, they may well indirectly manifest themselves in the literature when other scholars, such as Clarke (2010) or Ryan and O'Connor (2008), discuss such things as learning environments and how they are nurtured through iterative processes and collaborative efforts. To summarize, this literature review will assess or capture the success of the case study by seeing to what extent it meets the criterion of an open and robust learning environment in the context of a software development project and its knowledge exchange within as conceptualized by Nonaka and Konno (1998) and further turning it into a model.

Engaging with the literature above may prove very powerful and efficacious in a host of ways. It will highlight the extent to which the events in the case study coincide with the literature insofar as corroborating key findings vis-à-vis the efficacy of specific practices. Put another way, do any practices championed in the literature, to the extent they are applied in the

case study, actually produce the benefits their authors insist they do? In bringing to a close this section, one may surmise that the literature will inevitably provide many useful extensions for the organization when it battles to bolster knowledge transfer and to facilitate optimal performance and hence success.

Within this chapter, different concepts were explored in detail which are the building blocks for the data evaluation and later the construction of the model. The theories of Nonaka and Teece (2001), Clarke (2010), Ryan (2012) as well as the NDM model complement each other in the coming chapters, where each theory will help evaluate different aspects of the data. In addition, they will hinge upon the results and drawn conclusions. Having discussed tacit and explicit knowledge in a broader context as well as project management and software development in chapter two, In the following chapter the focus will be the research methodologies used to gain data and the process of data evaluation.

Chapter 3: Methodology

3.1 Introduction

In the previous chapters, research questions evolved from an analysis of the literature set against a reassessment of the research aims. In this chapter the methodology used to answer these questions as well as how the data was collected will be presented and discussed. Below the questions of the research are listed and further explained.

1. What is the current understanding of tacit and explicit knowledge exchange in IT Software Project development?

The first research question aims to critically examine concepts and theories used in the field of tacit knowledge, in particular when applied to software development projects. This builds upon an understanding of the current literature and focusses on more current theories. This allows a theoretical outline which aids the data collection as well as evaluation. These theories also provide the basis for addressing research questions 2, 3 and 4.

2. How are tacit and explicit knowledge exchanged in software development projects?

Question 2 focuses on collected data and its evaluation, aiming to show the flow of knowledge. More specifically, the data should demonstrate the exchange of tacit and explicit knowledge exchanged during a software development project through the theories assessed in answering question 1 utilizing the methodology introduced in this chapter.

3. How does the evolution of tacit and explicit knowledge in a software development project affect individual and group decision making and outcomes?

Question 3 further evaluates the data from question 2, using the NDM model it is aimed to filter out decisions made during the meetings. This aims to show how decisions are made and how other team members' tacit knowledge impacts the decision making of the project.

4. Can tacit and explicit knowledge be recognized and harnessed by a conceptual model in software development projects?

Finally, question 4 uses theories developed through researching question 1 and their usage in research questions 2 and 3 to formulate a new model. The construction of the model is a combination of several models which are used to analyse the data and then later reflected within.

The study used an exploratory, in-depth, participant observation technique to compile the data over a three-month period. Meetings were recorded, where the three companies discussed the amendments needed to complete the project. Wijetunge (2012) argues that tonal cues enrich the story with voice fluctuations than written stories. The aim was to establish a viable software tailored to the needs of the customer.

In the following sections the research and methodological approaches will be discussed. Research procedures and project elements, such as people, companies and the software discussed, will be explained. The data collection process is discussed, how it was evaluated as well as their relationship to the research questions. The researcher's perspectives, ethical considerations and a summary finalize the chapter.

3.2 Research Approach

The research approach is centred around an in-depth, exploratory investigation, which seeks to reveal tacit and explicit knowledge in a software development environment during its creation and augmentation, from an individual as well as a group perspective. The key aim of the study is to demonstrate how project teams in software development create and exchange tacit knowledge explicitly during social interaction over a period of time. In contrast to most of the research being based on participant's recall of tacit knowledge and their personal interpretation, this study seeks to expose tacit knowledge during project meetings, hence at the point of transformation from tacit to explicit.

To explore the meaning and purpose of a conversation between project members, a contextual methodology is needed. Understanding context is crucial for the comprehension of a project, the decision making and knowledge input process of team members involved. The triggers and cues utilized to transform tacit into explicit knowledge during a project meeting needs a “broadly interpretive method of research” (Walsham, 1995) in order to surface. An interpretive approach lures on the words and experiences exchanged by participants as well as the researcher’s interpretation of the exchanged knowledge. This hinges on the epistemological belief that social belief is directly related to the interpretation of meaning during a specific phenomena and those interacting with them (Orlikowski and Baroudi, 1991). The purpose of interpretive studies is to research a specific situation and its context in order to gain a greater understanding of the phenomena. Unlike positivist approaches, the aim is not to seek a generalized perspective but rather to inform others of a specific setting (Orlikowski and Baroudi, 1991; Walsham, 1995). A contextual, interpretivist methodology is therefore applied.

In order to select an appropriate research technique, the requirements of the research need to be addressed. There are three major issues to deal with when assessing the research. To begin, an exploratory research approach to analyse the situation was conducted, as the research went along it became more inductive. This requires a reasoning which hinges on participant’s social interactions and the interpretation and judgment of the researcher. The assessment of a situation with pre-determined questions can limit the field of enquiry rather than permitting the researcher to understand a complex behaviour without imposing the risk of categorization (Punch, 1998). Tacit knowledge is learned by experience and not commonly known (Sternberg et al., 2000; Sternberg, Horvath, 1999), therefore in attempting to draw out tacit and explicit knowledge in a project, the nature of creation and exchange needs to prosper, while restraining to influence the topics discussed or decisions made within.

Secondly, the research focuses on exploring how tacit knowledge is exchanged during a project. Welman and Kruger (1999) state that “the

phenomenologists are concerned with understanding social and psychological phenomena from the perspectives of people involved”, hence a phenomenological approach to the evaluation of the study is needed. The moment of tacit knowledge creation and its direct exchange in the project team is one of the main goals of the research. It is imperative to evaluate the given situation from a group or individual point of view rather than asking them to recall situations or feelings about a past project. According to Berry and Dienes (1993) tacit knowledge is mainly context-specific and is difficult to transfer to tasks with a different context as well as actions taken by participants can differ from actual practice (Argyris and Schön, 1978). The project team’s context in which the tacit and explicit knowledge is created is harnessed on an individual or group perspective and their terms (Denzin, 1970; Robertson and Boyle, 1984). Tacit and explicit knowledge can therefore be directly extracted from the moment it is created and altered.

Keeping in mind that “inquiry doesn’t mean you are looking for answers” (Kabat-Zinn, 1986), a combination of methodologies is aimed at meeting the requirements above. In the coming section, an outline of the research procedures is described with in an-depth look at the companies as well as actors involved, followed by an assessment of participant observation, unstructured interviewing, and the critical decision making model.

3.3 Methodological Approach

Hycner (1999) argues that a “phenomenon dictates the method and not vice-versa.” The methodological approaches explained in this section are built around the collected data, with the aim of extracting tacit knowledge through the perspectives of the participants. The study sets out to explore the phenomena of tacit knowledge in software development projects and its transformation into explicit knowledge. A contextual, narrative approach is needed to extract tacit knowledge due to its intangible nature. First, a

discussion of phenomena will set the basis of the methodologies, followed by participant observation and finally the use of unstructured interviews.

First, a phenomenology approach was used in the study, which allows the inquisition of a specific situation from the participant's perspective (Welman and Kruger, 1999). The method was first specified in the 19th century with Edmund Husserl, who's aim was "to develop a new philosophical method which would lend absolute certainty to a disintegrating civilization" (Eagleton, 1983). Martin Heidegger, Husserl's student, created the concept of "Dasein" or "Being there", which assessed a person's dialogue with herself and her world. The study of phenomenology was not accepted for a long time in the natural scientific community as a viable alternative to traditional methods due to its lack of a systematic approach (Stones, 1988). By the 1970s, phenomenological psychologists established "a methodological realisation of phenomenological philosophical attitude" (Stones, 1988).

The aim of phenomenology to a researcher is its concern for the lived experience of people (Greene, 1997; Holloway, 1997; Kruger, 1988; Kvale, 1996; Maypole & Davies, 2001; Robinson & Reed, 1998) and its description. Benz and Shapiro (1998) elaborate that phenomena are to be understood in their own terms, hence a description of the human experience as it occurs and from the perspective of the person researching it (Cameron, Schaffer and Hyeon-Ae, 2001). Using a qualitative approach such as phenomenology therefore implies the use of a specific phenomenon. The research approach was to allow a project to evolve and unravel itself naturally while observing the participants in their actions and interactions with each other.

Participant observation is an essential part of phenomenology and often referred to as a general approach of fieldwork (Spradley, 1980). To scholars such as Agar (1996) the definition of participant observation is a general approach to the observation of formal and informal interviewing in which anthropologist engage. As a methodology, it entails the observer taking part in rituals, daily activities, interactions, and events of people being studied as one of the means of learning the explicit and tacit aspects of their

culture in this study (Dewalt, 2010). The method of participant observation embraces the goal to understand tacit knowledge at its creation, while embracing the cultural aspects of a software development project.

Using the OPPTY approach (Leonard, 2013) allows the project team to learn from an expert within the group. This can be seen during the learning software models of the software. By observing a team member performing a specific task within the software, the team then practices and mirrors the behaviour of the mentor. Here feedback is given by the teacher. Once the team members have understood a specific page or process within the software the teacher and the other team members partner up and work together to address the challenges and opportunities within the software. Finally, the team takes responsibility and uses the knowledge gained from the teacher within the team. This then allows the knowledge to be combined and used by the project team with their own expert knowledge.

To find a balance between participation and observation, the terms first need to be defined. Participation according to Benjamin Paul (1953) implies an emotional involvement, and further described as “going native” or becoming the phenomenon” (Jorgenson, 1989). In contrast, observation requires detachment and seeks to remove the researcher from the actions and behaviours so they are unable to influence them (Dewalt, 2010). During the study, the research approach was a “participant observer”, where the researcher is known to the group as an observer, but also participating in activities (Robson, 2011). An apprenticeship approach, unlike less participatory approaches, allows a researcher to experience results in “ways of knowing” and “learning to see” (Coy, 1989). Clifford (1997) argues that it will be more difficult for a detached observer to examine research assumptions and belief as well as themselves than for researcher engaging in participant observation.

Field notes are an essential part of participant observation, since observations are not data unless they are recorded in some fashion for further analysis (Dewalt, 2010). Seligman (1951) suggest that there are three main categories of field notes, 1) records of events observed, a method

where the researcher converses or interviews participants during the events, 2) records of prolonged activities or ceremonies, where an interview is not feasible, and 3) a daily journal. Seligman (1951) suggest that “the investigator must sense the native attitude to note taking in public”. This research focused on assessing a prolonged activity / ceremony, a weekly meeting for three months to achieve a feasible and functioning human resource software. These were recorded as voice memos for further investigation after the field work. A journal was also kept with feelings and events towards the research. However, the practice of taking notes in front of participants can be uncomfortable and objectifying (Jackson, 1990). This was highlighted in the meeting of the 3rd of April, where one of the participants was curious about notes taken during the meeting, however it was pointed out by the researcher that the notes were for comprehension of the software.

“I am terribly worried about what SD C is writing down.” – SD A

“I am trying to understand the tools as much as you guys (HR A and HR B) are.” – SD C

When commencing a field study one of the first steps is to produce a rapport with the subjects. She (1986) defines a rapport to be the moment where the anthropologist and his informants establish a line of communication in order for the former to collect data which allows the understanding of the culture under study. While it is difficult to establish a rapport in many areas of field work, in the research at hand, this line of communication is established through the weekly meetings of the project team. The role of a researcher is to learn, therefore observing the project members and understanding the topics discussed is a natural part of “job”. The subjects of work discussed during the meetings come up naturally, since the issues of the project need to be discussed and resolved to achieve project success, hence topics and related queries emerge naturally without the researcher interfering. In the following sub-section the unstructured interview approach will be further discussed.

Using complete participation, apprenticeship, entails the acknowledgement of the researcher in respect to her perspective of the

research conducted throughout the course of the study as well as reflecting and acting on research ethics. This will be further discussed in the sub-sections “researcher’s perspective’ and ‘ethical considerations”.

3.3.1 Grounded Theory

Grounded theory strives to understand the experiences of people in a rigorous and detailed manner (Ryan and Bernard, 2000). Concepts and categories that emerge from textual analysis are linked together to be then associated with substantive and formal theories. The phenomenon studied by a grounded theorist greatly relies on the intensity and habitual process of analysing the data and progressively becomes more ‘grounded’ in the study. During this process concepts and models are being developed and applied to further understand the phenomenon. Transcribing and listening to interviews allows a basis of analysis, which are then processed into small samples of text to be read line by line. Sadelowski (1995) argues that the proofreading of the material is the beginning of analysis since at this point it makes some “yet inchoate sense”.

Followed by the initial proofreading, the data needs to be coded, which in grounded theory is often referred to as “open coding” (Agar, 1996; Ryan and Bernard, 2000). Open coding is the process of identifying potential themes extracted from real examples from the text. Progressively more categories and themes evolve through the analysis to then be linked with theoretical models gained through the literature. There are several methods in analysing the material from a grounded theory point of view. One is “constant comparison method” (Glazer and Strauss, 1967) where themes and concepts are compared and contrasted while asking questions such as “when, why and under what conditions do these themes occur.” Then, the method of ‘memoing’ uses note taking to describe discovered concepts, summaries of what is going on in the text from the researcher’s point of view and notes about practical matters. Strauss and Corbin (1990) describe these as code notes, theory notes and operational notes. Derived from the chosen

model, grounded theorists frequently display their results by presenting segments of text, where quotes from participants are demonstrated, as an example of concepts and theories (Ryan and Bernard, 2010).

3.3.2 Coding

Codes underline or aid in the identification of specific themes within a text. The researcher's aim when coding is to "identify the range and salience of key items and concepts, discover relationships among these items and concepts, and to build and test models linking these concepts together" (Ran and Bernard, 2000). Once the texts have been inductively analyzed, the next step is to find specific themes within the data through coding. This can be done through a codebook, which according to MacQueen (1998) includes a detailed description of each code, text extractions from the used data as well as in-and exclusion criteria. Developing and refining codes in texts is central when seeking categories. This can be done through marking text extracts, which is a vital process in codebook refinement (Krippendorff, 1980).

Marking is a way to underline codes within a text, Ryan and Bernard (2000) state that "the act of coding involves the assigning of codes in contiguous units of text. They further argue that codes act as tags which examine text through marking to be later retrieved or indexed. This can be in done for larger extracts of texts such as phrases but can also continue on for several pages, therefore they are not fixed units. A correspondence to reliability or dependability, needs to be demonstrated in order to verify findings. Transparency, to verify the findings, can be achieved through the use of leaving an audit trail or a codebook which sheds light to the research process and the reasoning behind conclusions made. This method entails the use of the researcher's range of knowledge in generalizing and accommodating comparable context to evaluate similarities and differences (Johnson et al., 2006).

3.3.3 Modelling and Displaying Concepts

Modelling and displaying concepts is a common tool used in qualitative research, it aids to represent relationship between themes or concepts. Using visual aids support the researcher during the analysis and help readers understand the chain of thought. Influential ways to communicate ideas include key quotes as examples, building forms or matrices, models as well as flowcharts or maps to represent theories (Ryan and Bernard, 2000).

Tables are a great method to support the coding process of text. The organization of raw text through tables enables the researcher to display or summarize data in multiple dimensions by filling out celling with quotes (Bernard and Ashton-Voyoucalos, 1976). Models are another way of showing qualitative data, often represented in forms of boxes - used for themes - and arrows - used to represent relationships among them. These relationships can be of different natures such as time, choices or association.

3.3.4 Ethnographic Decision Making

Ethnographic decision making models (EDM) is a qualitative approach to analyse behavioural choice in a specific context. This is one of the main methods of research. EDM's are decision making models which are based on if-then statements which link criteria and yes or no types of behaviours. These are often displayed in forms of decision tables, set of rules in an if-then statements or decision trees (Ryan and Bernard, 2000). These help to categorize statements made by participants to aggregate a decision making process.

3.4 Organisations, People and the Project

3.4.1 *The Companies and People Involved*

The following section will describe the field of research, focusing on the people and companies involved. The project is aimed to develop a cloud, internet, based human resource software for a small to medium sized enterprise in the United Kingdom. The project consisted of three different companies working together to organize and develop a software in accordance to UK human resource standards as well as the customer's needs. The research focused on the software development organization's interplay with the human resource experts and the customer. In the following section the three companies will be presented as well as the actors of the software development project. Finally, the connection between the three companies will be discussed.

The software development organization is a solution provider for a cloud based ERP (Enterprise-Resource-Planning), specializing in human resource and accountancy ERP. Focusing on selling NetSuite products, the organization started to specialize in the further development of solution applications. The organization consist of four permanent employees, a chartered accountant, an administrator and the software developer, there are several employees which come on a project basis. Through the in-house software developers, the software development organization was able to customize a human resource program for the human resource consultancy.

The second organization is the human resource consultancy, specializing in SME (Small to Medium sized Enterprises) human resource management and employment law. Their customers usually outsource their human resource division to them, taking care of the human resource administrative tasks as well as payroll. The current system used by the human resource consultancy was outdated and most of the tasks had to be done manually. Trying to facilitate work processes for them as well as their clients, the human resource consultancy turned to the software development organization, to develop an online human resource program customized to their specifications and requirements.

Finally, the customer of the software is an independent charitable organization specializing in heritage advice and services. Working in development, infrastructure and construction the organization aims to store and protect archaeological artefacts. The organization has a core staff of 250 employees based in London, however a large amount of their staff are 'bank staff', project based archaeologists hired for a specific task. These archaeologists work throughout the United Kingdom as well as internationally and are part of the human resource data base but are not permanent staff. This is one of the great challenges the HR software faces and greatly influences the demographic accessibility as well as the constant change of current staff. The customer being one of the largest clients of the human resource consultancy, the software was tailored around their needs.

Throughout the project there were twelve people mainly involved in the development of the human resource software. Below the main actors of the project are discussed.

The software development organization consists of 5 main employees involved in the project. The first, software developer A, SD A, is a consultant and occasional programmer. SD A mainly focuses on consultancy, but also makes minor changes within the software. He / She is in charge of major decisions of the software and has general programming knowledge, however the expertise lie within accountancy. SD F is an administrator who takes care of the administrative tasks, such as negotiations with the customers as well as smaller tasks such as scheduling. Following, SD B is a programmer and consultant. She is head programmer and functions as a consultant at times. Mainly she focuses on the infrastructure of the program and makes all the major programming decisions. SD D is a programmer who supports SD B. Finally, SD C is a researcher and trainee. He / She is primarily a researcher, but also functions as SD A's trainee, and supports her as well as SD B in various tasks such as testing the software. He / She takes part in all of the human resource consultancy and customer meetings as a researcher.

The human resource consultancy (HR) also has five main actors in the project. HR D is the executive of the HR consultancy. She is a key decision

maker and works on an executive level. Her main goal is to understand the progression in the project and make sure there is a positive outcome. HR A is an HR manager. She is one of the key Figures in designing the software and focuses mainly on human resource law as well as administration. HR B is also an HR Manager. Like HR A she is one of the key Figures in designing the software and focuses mainly on payroll and human resource administration. HR C is in charge of recruitment for The human resource consultancy and aids in the development of the recruitment option of the software. Finally, CL B is the accountant who takes care of payroll for The human resource consultancy.

The client (CL) only has two main people involved in the development of the project. CL A, the head of human resources. She advises the human resource consultancy and the software development organization in topics regarding the customer in practices and expectations of the software. CL C is CL As right hand and also in charge of payroll. Later in the project, the employees of the customers are trained in the software, however they did not take part in the development process of the software.

3.4.2 The Project

The three companies have a common goal, however their approaches and individual aims differ. The success of an entrepreneur and their organization stands in direct relation to their ability to manage an innovative product well and thus positively affecting the lives of others (Agbim and Oriarewo, 2012). With this in mind, the companies undertake the goal to support each other in reach project success and achieve individual goals.

The software development organization are specialized in accounting payroll and have acted as a vendor of NetSuite since the 90s. They seek to grow the organization and offer their customers a larger variety of software solutions. The human resource software gives them an opportunity to sell a new solution, through the help of The human resource consultancy they can

use applied human resource expertise to shape the software. The customer gives them a platform to test the new software and a first reference.

The human resource consultancy have had several issues with the customer due to their failing human resource system and needed a new solution to satisfy their customer. In addition, it being a tailored software in accordance to their needs, the solution can facilitate their work and help a more efficient working climate for the customer as well as other clients in the future.

The customer needs an HR software to facilitate some of the processes within the organization. Most of the processes were done manually by the team, such as absences, payroll or learning and development opportunities. Another issue is the need to reach a wide demographic audience, due to not all of their staff not working in house. Their aim is to have a tool which is easy to use for their employees and at the same time gives them the opportunity to enter their HR needs in one place as well as facilitating internal HR processes.

3.5 Data Collection and Analysis

3.5.1 Collection

The data collection was done over a three-month period, where weekly meetings were held for the project. Creswell (1998) states that long interviews with up to 10 people, Boyd (2001) states two to ten participants, is sufficient to explore a phenomenological study which fits with the researched group of twelve people, where five people take a key position in the project are the focus of the study. In total 34 hours of meetings were recorded which centres around the refinement of the software. The exchange of tacit and explicit, expert, knowledge during this period was elevated due to the participants aim to finalize the software for operation and trying to “get it right” under time pressure.

The project was limited to the 34 hours, since the aim of the research was to focus on the project before going 'live'. The beginning of the project started with the weekly meetings where the participants sat together and could exchange their expert knowledge among one another. The data collection ended with the project going live. Using grounded theory, participant observation allowed a more dynamic approach to the research. The two main sources of data were recording meetings and field notes. Once the data was evaluated it became clear that the field notes were not revealing the participant's tacit knowledge exchange but merely personal thoughts of the researcher towards the different actors and their role within the project. This however could also be seen through the recorded meetings. Emerging through the richness of the recorded meetings, the evaluation of knowledge being passed on from one participant to the other and their oral response became the focus of the research.

The participants of the study were predetermined by the project leaders. Kruger (1988) states participants need to have had experiences relating to the phenomenon to be researched. The different expertise offered by the participants in the project are prerequisite to their involvement in the project. The different fields of work involved in the project, require expertise from the human resources, software engineering, consultancy and finance. The participants are discussed the section companies and people involved.

The data collection ended at the launch of the project, after the training sessions were held, due to the project officially being over and the weekly meetings finishing as well. After this period, the focus lies constant optimization of the software, which is operational and not part of the development project. Therefore, follow up – recall – interviews were not held since the aim of the research was to capture tacit knowledge at its creation and augmentation during the development phase, not revisiting the project.

The unstructured interviews come in form of meetings, where the natural flow of conversation between project members permits tacit knowledge to unfold and allows its transformation to explicit knowledge. As previously stated, the form participant observation – fieldwork – is the

recording of a prolonged activity or ceremony. Throughout the meetings, the goal is to resolve issues and to fill participants' gaps of knowledge in order to achieve a well-functioning, complete software. In the following section, methods to analyse text in a narrative form are discussed as well as specific models used to answer research questions 2 and 3.

3.5.2 Analysis

34 hours of meetings were recorded over the three-month period. The meetings were each set out to last three hours twice a week, with a conference call between a human resource consultancy organization, a software development organization as well as a customer in need of a new human resource software, once a week. However, due to holidays and several hours of data input into the system some of the sessions were not taken into account. The group agreed on the research and recording of the meetings during the first meeting. At this point the role as a researcher in the project was established. The meetings could run short as well as over time and some included data entry into the software, which is a purely tacit action and therefore does not have an audio trail. At the very end of the project, training modules were held on-site at the customer's headquarters, which were also recorded. These different meeting approaches needed to be considered during the analysis. This subsection will discuss the analysis procedure as well as methods used to analyse the data.

The meetings consisted of three stakeholders, the software development organization, the customer and a human resource consultancy firm. Within each organization different roles can be found which contribute knowledge to the project. The software development organization has software developers or engineers, who work mainly in the back office, who at times communicated with the human resource organization or the customer when necessary. Their knowledge was software specific and needed input from the software development executives or managers, who advised them on the customer's needs. These managers are crucial for the interaction of

the customer's needs and the software developer's capabilities. The human resource consultants played a vital role in shaping the software development manager. In addition, most decisions were made by the human resource consultants and the software managers according to human resource standards and the customer's needs. The executives from the 3 companies were essential for the decision making over time.

The 3 cycles

The collected data from the meetings were analysed in 3 cycles. During this period themes and theories emerged. The inductive approach allowed theories from several researchers to be used and matched with the found data. However, seeking tacit knowledge and setting out with the basic theories of Nonaka. In the following section the 3 cycles are explained. Followed by the main theories used to analyse the data and finally the results of the data analysis which focuses on tacit knowledge triggers.

The first cycle of the data analysis was categorization of the recordings according to date and time as well as calculating the total hours spent in meetings. This gave an initial overview of the time needed to analyse the data. Using the journal, the topics discussed in each recording could be compared and matched accordingly. Miles and Huberman (1994) argues that to start a research some general themes derived from the reading of the literature should be used and progressively throughout the study more should be added. A general frame within the tacit knowledge spectrum was established before commencing the research. Exchanged tacit and explicit knowledge were taken into account as well as trigger points studied in the literature started out this research. Using a table approach, the meetings were categorized by the date, within each spreadsheet there were nine columns: date, duration of the meeting, general (topic of discussion), detail (details about the discussed topic), comments (thoughts of the researcher), general action (what was the aim of the discussion), tacit knowledge (explicit to tacit, tacit to explicit, trigger point) and the outcome for

the others (who learns?). The aim is to discover relationships among items and concepts (Ryan and Bernard, 2000), the first round of coding allowed a more detailed overview, which could then be intensified and elaborated during the second phase.

During the second cycles of analysis, precise conversations were transcribed. The aim is to assess the moments of tacit knowledge exchange, creation and triggers which emerge from social interaction. The conversations were written down in tables, however this time the spreadsheets were categorized by the general subject of discussion. This allows a more precise form of analysis of the conversations which allowed a closer look at the narratives over time. Once again the spreadsheet had 9 columns consisting of general information such as the date and duration (at times there was more than one recording a day), this helps finding the right recording for future reference) and the environment (i.e. conference call, briefing, learning software). The fourth column was the person speaking, followed by the addressed subject and sub-subject (i.e. payroll - additions and deductions – pay by period) and the example / quote from the person. The final two columns were categorized in trigger points and internal and external (i.e. tacit to explicit or explicit to tacit). Spradley (1979) argues that the intent of text analysis is to find evidence of social conflict and information about problem solving. The two rounds of analysis support a developmental analysis of group and individual tacit knowledge augmentation within a specific topic and is aimed to help the analysis of how the problems were solved, where there was conflict and how people used their expert knowledge. Ryan and Bernard (2000) say that a researcher has identified the themes and refined them to the point where they can be applied to an entire corpus of texts, a lot of interpretive analysis has already been done.

The third cycle used the emergent themes from the first and second cycle and analysed them through text analysis to find themes and their relationship to theories. These were then used to answer the second research question – exploring tacit knowledge –, the third – the impact of tacit knowledge on decision making – as well as the fourth – recognizing and harnessing tacit knowledge in a conceptual model.

1. Research Question – What is the current understanding of tacit and explicit knowledge exchange in IT Software Project development?

Research Question 1 was answered in the course of the second chapter. These chapters allowed an overview of the current theories of tacit knowledge in software development projects. A systematic analysis of current literature was conducted. Theories found during the analysis allowed the further development of the following research questions.

2. Research Question – How is tacit and explicit knowledge exchanged in software development projects?

The analysis of the second research questions started with a categorization of topics discussed during the meetings. These were then put into order of date and time, to see how topics, such as triggers evolve and develop over time. During the analysis the main focus was based on the concepts which emerged during the literature, which answered the first research question, the current understanding of tacit and explicit knowledge in software development projects. The focus lies in the comparison of the transcriptions to the SECI model, TTKM as well as the tacit knowledge spectrum. Through a narrative transcription constructive learning, socialization and externalization are highlighted. New group tacit knowledge emerges when internalization from individuals take place. Finally, the internalization process is assessed in more detail in order to understand tacit trigger points. These triggers emerge from social interaction, constructive learning or external influences. The co-dependent nature of the theories allows a systematic breakdown of the conversation. The origins of the surfacing tacit knowledge, and its build up through the conversations can be evaluated.

In the analysis, the examples are categorized by the topic of discussion, i.e. finance, recruitment or 360-feedback, followed by date and time of the meeting. This order aims to show how knowledge builds up, is exchanged and transferred over time. In addition, the different parties involved in a topic can be evaluated and its transfer from one team member or group to the next will later be assessed. After each section two tables will sum up the tacit knowledge found in each extract. The first table's focus lies on the SECI, group tacit knowledge and constructive learning. The second shows what kind of trigger caused the tacit knowledge to surface, these categories were found through literature as well as during the data analysis phase. The trigger tables, Table 3 and Table 4, will be discussed in more detail in the results chapter.

Table 3 -Example Table Tacit Knowledge Data

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning

Table 4 - Example Table Tacit Knowledge Triggers Data

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger

Names are changed and sorted by organization, SD – software developer, HR – human recourse consultants and CL – client, AC - accountant and then the team members name was changed by a letter A – Z. The organization upgrading their HR software, all old software used by the organization are referred to as 'old software' vs 'new software". Below is a list of the people involved in the project and their role.

Software Development Organization:

SD A: software development consultant.

SD B: software engineer.

SD C: researcher.

SD D: software engineer.

SD E: HR consultant for software development organization.

SD F: software development consultant.

Human Resource Organization:

HR A: human resource consultant.

HR B: human resource consultant.

HR C: recruitment specialist.

HR D: head of human recourse organization.

Customer:

CL A: customer head of HR.

CL B: accounting employee of customer.

CL C: HR employee of customer.

AC A: Accountant

AC B: Accountant

The greatest challenge of the data analysis was to filter out tacit knowledge and what triggers knowledge to surface. The meetings were all evaluated through recorded audio tapes, where the main goal was to find tacit knowledge surface. The unstructured nature of the meetings gave way to tacit knowledge surfacing from each individual when it was necessary for the project or felt important to a team member. Topics could change due to an external factor, trigger, when it was required by the team or subject at hand. Not searching for specific words within the conversations but rather understanding and evaluating concepts or questions communicated to other

individuals required a large amount of inductive research.

Filtering out when constructive learning becomes social interaction and vice versa was the first step was simpler to evaluate. Evaluating internalization posed a great challenge, due to the process being internal. This could be evaluated through a constructive response to either constructive learning, socialization or external factors such as visual components such as a page in the system (i.e. a page within the software displaying the information of an employee). During the analysis process several triggers emerged which allowed tacit knowledge to surface, highlighting them in the discussions was difficult. This stage of the analysis is the most complex and vital to find answers to the research questions, since it builds the basis of data for the third and fourth research question. The extensive collected data needed to show knowledge being transferred from one team member to the next. The time component and the way knowledge is then used to understand decisions the decision making process.

3. Research Question - How does the evolution of tacit and explicit knowledge in a software development project affect individual and group decision making and outcomes?

The decision-making process was analysed through the meetings, where an evaluation of conversations was made. It was evaluated where decisions were made and when the decisions arose again at a later stage. At times decisions which were forgotten surfaced again through a process of recollection. These extracts were then analysed through the naturalistic decision making scheme, which was previously evaluated in the literature. The focus of the decision making during the meetings, were the individual and the group influences on making decisions, i.e. when another project member was needed to make a decision or when a project member recalled a decision previously made by the group. The naturalistic decision making scheme aids in evaluating decision making within the group and at times on an individual level.

4. Research Question – Can tacit and explicit knowledge be recognized and harnessed by a conceptual model in software development projects?

Research question 4 combines the three previous questions in theory and data. The emerged theories during research question 1 allowed the evaluation of research question 2 and 3 through models proposed by Nonaka, Ryan and Clarke, as well as the decision making tree. During the analysis of research question 2 all models were needed to fully analyse and understand tacit knowledge in the meetings. Nonaka gave a basis to understand the dynamic environment the face-to-face and conference calls enabled. Furthermore, SECI gave general areas of tacit knowledge allowing the spiralling tacit knowledge to build up. Ryan's model gave a more detailed view of the group process of acquiring tacit knowledge in a software environment. Finally, Clarke demonstrates the internal process of tacit knowledge acquisition, which Nonaka defines as internalization and Ryan as individual knowledge. The trigger found in Clarke's model is internal and occurs after knowledge input and a round of reflection. During the research several different triggers surfaced, which are explained and evaluated in answering research question 2.

The models were then analysed in accordance with the found data in research question 2. They were then put back together in accordance to the data found and flows of tacit knowledge which surfaced over time. Using different approaches to tacit knowledge in teams and individuals allows a more complete view of the process.

3.6 Researcher's Perspective

Commencing as a novice researcher, the aim was to learn and understand the software in accordance with the progression of the project as an observer. Fully immersing one's self into the project was crucial to later

understand actions taken as well as understanding the progression taken the researcher took part in all events related to the project. Having the consent to research the project, the inclusion into the project as part of the project team helped the other participants to forget the primary role as a researcher.

Unlike positivist approaches, phenomenologist cannot be detached from their own presuppositions and should not pretend otherwise (Hammersley, 2000). A completely unbiased research when utilising phenomenology is hence not possible and the researcher is bound to hold individual explicit beliefs (Mouton and Marais, 1990). In addition, Creswell (1994), Manson (1996) and Holloway (1997) argue that a researcher's epistemology is literally within her theory of knowledge and the decision how the phenomena will be studied. During this study, the approach to the research was to collect data which is contained within the perspectives of people involved in a software development project and to observe and engage with participants related to this phenomenon.

3.7 Ethical Considerations

For confidentiality reasons, the real names of participants of the study and the organisations are not used. All of the people involved were reassured that their participation in the study was voluntary and assured that their data would not be used in the study, unless anonymised, without penalty. All participants of the study were informed of the nature of the research as well as the researcher's role of apprentice.

The aim of the research was to investigate tacit knowledge in a software development meeting context. An inductive research approach allows a data set to be interpreted to generate meaning to find patterns and relationships (Bernard, 2011). The inductive research approach, however does rely on pre-existing theories (Saunders et al., 2012).

This research approach comes with ethical considerations such as the researcher's bias towards the interpretation of the data. Nickerson (1998)

argues that confirmation bias leads a researcher to seek or interpret materials with their partial pre-existing beliefs, hypothesis or expectations. It can therefore be said that the research was set out to find the theory of tacit knowledge, however the interpretation is set on existing theories.

In addition, due to the data being gathered in a business, the material such as names were kept confidential. The participants were recorded and made aware of their role within the study, as well as the participation being voluntary.

3.8 Summary

This chapter discusses methodologies used for analysis and their utilization in the research. The key issues and research approach was presented. Methodologies used to approach these issues as well as the research were then presented in order to gain an understanding of what the researcher set out to do. Following, research procedures, with an in-depth look at the companies and participants used for the study were examined. Being a participant observer, the goal of the methodology was to be able to ask questions and understand the group dynamics as well as the software. Evaluating the data through a qualitative approach allows a narrative evaluation of the meetings. Drilling down on the collected data through several rounds of coding the aim is to filter out when tacit knowledge emerges and through which channels. The data collection procedures and its analysis linked previously discussed methodologies and elaborated on precise research techniques. Research questions and their relationship to collected data and its evaluation was reviewed. To conclude this section, the researcher's perspective and ethical considerations were discussed. The following section will discuss the data analysis.

Chapter 4: Concepts, Theories and Results

4.1 Introduction

The data analysis mainly focused on finding evidence of tacit knowledge. During the data collection and evaluation different concepts of tacit knowledge were used for the analysis process. First, the basic literature of tacit knowledge gave a foundation for the evaluation. These concepts and theories were refined once the data was analysed in more detail during the research cycles. These different theories emerged through the data.

In this chapter, the main results of the research cycles will be discussed, followed by the theories used to evaluate the data. These main theories focus on the types of tacit knowledge exchange and use approaches from Nonaka and Teece (1998), Ryan (2013) and Clarke (2010). These are later analysed in the following chapter.

4.2 Results from the 3 Research Cycles

4.2.1 Research Cycle 1

The first research cycle focused on the categorization of the recorded data. The aim was to find indications of tacit knowledge in respect of the theories previously examined. The focus was tacit knowledge transfer within the project. During this cycle, no meetings were transcribed but merely listened to. As previously discussed in the methodology, the hours of the meetings were recorded along with the main purpose of each meeting. This was done in accordance with the field notes. In addition, personal comments, the general knowledge transfer action, as well as the outcome of the other participants (i.e. learning, teaching, resolving a problem) were categorized.

The first cycle of research analysis allowed a broad overview of the material. It gave the research an outline and provided indicators toward which area of tacit knowledge the research should focus on. Finding evidence of tacit knowledge allowed further investigation into the material. At

the beginning, knowledge was simply categorized into tacit knowledge, explicit knowledge, tacit knowledge converted to explicit knowledge and tacit to explicit knowledge. In addition, group TK (group tacit knowledge) was categorized as an outcome for the project team. Other findings such as the recorded behaviours of the participants did not have relevance in the following research cycles. These did not prove to have any relation to tacit knowledge within the research scope. However, general actions such as learning or discussing the software and its issues played a vital role in the subsequent research cycles.

The knowledge within the project was spread between the different companies, and team work was needed to achieve success. Füller et al. (2015) argues that customer integration into new product development strengthens the core company competencies. In this research cycle the data was laid out and analysed to help understand the interplay between the companies which can be seen in Figure 11.

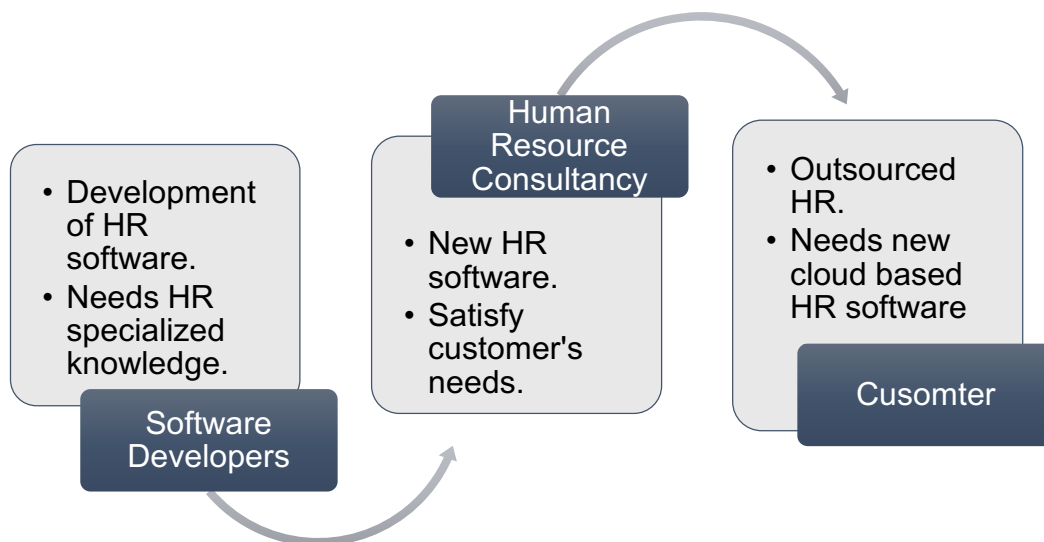


Figure 11 - Companies involved in the Project and their needs

The software developer's goal is to engineer an HR software which can be used and sold to several different customers. The aim is to construct a complete HR software containing all facets of HR management. This

includes topics such as payroll, employee management, recruitment and disciplinary and grievances. They are focused on getting as much expert knowledge from the HR consultants as possible to construct a well-functioning and complete software tailored to the needs of human resource employees.

The human resource consultant's project goal is to obtain a human resource software which satisfies the customer's needs. Therefore, a well-functioning and complete software is also in their interest. They also aim to use the software for other customers, therefore the efficient usage of the software for their daily work is of great importance to them.

The customer outsources their HR department and want a cloud based solution. A cloud based solution allows their employees to access their employee records from any computer. Having a vast amount of bank staff, it is the most efficient and cost effective way to manage their employees. Their goal for the software is for it to fit their specific needs and allows a more automated approach to human resource management.

The raw data outcome of the first cycle allowed general theories to emerge from the recordings. Finding a vast amount of evidence of tacit knowledge exchange as well as group tacit knowledge exchange, a deeper analysis of theories within the subject was made. Here the theories of Ryan (2013) and Clarke (2010) started to emerge. A more refined search of tacit knowledge continued in the second research cycle.

4.2.2 Research Cycle 2

The second research cycle focused on a more precise search of tacit knowledge within the recordings. The goal of the second cycle was to complete the transcription of the recordings. This allowed a more explicit analysis of the data as well as a more detailed analysis of tacit knowledge. At this point, the spreadsheets were no longer split into dates but into the topics of discussion (i.e. payroll, 360 feedback, disciplinary and grievances etc.). As stated in the methodology, the main categories of analysis were social

interaction, constructive learning, group tacit knowledge, individual knowledge, tacit knowledge triggers and decision making. Aiming to answer the research questions, the theories of Ryan (2013), Nonaka (1998) and Clarke (2010) were used for the data analysis. These were compared and contrasted in order to find theories which corresponded to the emerging data.

Companies share space and reinforce relationships between co-workers, demonstrating its foundation of knowledge creation. These relationships are formed in different scenarios throughout the work day. Some of the knowledge is formed through informal channels, such as a discussion during the coffee break, or formally through e-mails or meetings. When such a discussion occurs, may it be explicitly or tacitly, 'Ba' gives the basis for creation. This ongoing process, gives co-workers the ability to comprehend and combine knowledge in order to complete the task at hand. Establishing 'Ba' as the basis of the model enables a secure surrounding where knowledge can dynamically grow.

Flourishing through a dynamic shared space, knowledge spirals horizontally as well as vertically throughout an organization continuously (Nonaka and Teece, 2001). Socialization enables a tacit to tacit exchange, where experiences and know-how are shared and combined with others. This process can be inside or outside of an organization, and gives the basis for externalization, tacit to explicit (teaching), combination, explicit to explicit (gathering and combining knowledge from in-or outside of the organization), and internalization, explicit to tacit (learning). Focusing on Tacit knowledge, combination will not be part of the model, due to its purely explicit nature and will be referred to SEI model. SECI helps us view the general movements of knowledge creation and exchange within companies.

Establishing four general movements of knowledge exchange, a more in-depth examination is provided by Ryan (2013), through the team tacit knowledge measure. Using a quantitative approach Ryan demonstrates the movement of knowledge within a group and the moment of creation. Beginning at the current team tacit knowledge, constructive learning enhances individual knowledge, which can then again be shared with the

team in order to build up the transactive memory, ending in a new amplified team tacit knowledge. This new team knowledge then begins again in order to elevate the knowledge within the group in a never ending spiral of knowledge.

Clarke evaluates knowledge from an individual point of view and establishes a micro view on tacit knowledge creation. The model where reflection, triggers, tacit and explicit as well as existing knowledge take part in knowledge creation exhibits the flow of knowledge when received by a person within a team. In more detail the process starts with the receiver being fed with knowledge – knowledge input - which is then processed, enhanced and formed into a knowledge output. Clarke (2010) has addressed tacit knowledge triggers but did not categorize them nor looked at the channel through which they surface.

Combining Nonaka's theory of 'Ba', SECI and the spiral of knowledge gives a basis in order to understand the creation and general movement of knowledge. Ryan pushed the idea of knowledge creation within a group further and paves the way to further understand how knowledge is created and enhanced within teams. Lastly, Ryan gives an individual perspective of the flow of knowledge, which aids in the understanding how knowledge is processed within a person. Utilizing the theories of Nonaka, Ryan and Clarke a model demonstrating knowledge creation and movement within an organization on a group as well as individual basis can be created.

Ryan (2013) states that "individuals draw from the team tacit knowledge and create their own tacit knowledge. This is a background process which is dynamic and reciprocal relying on constructivist situated learning." This is the main focus of the analysis for research question 2. However, its reliance on constructive learning and social interaction influences constructive learning plays a vital role in the research.

The second research question is the most extensive in terms of data analysis. It later provides the basis to answer research questions 3 and 4. The analysis of question 2 uses theories found during the literature and conceptual framework. The main questions during the analysis are which

team members interact with each other and who carries what information. How does tacit and explicit knowledge spiral throughout the project team and when is there a new common understand of group or team tacit knowledge and finally when does individual knowledge surface and what triggers it.

Social interaction is the basis of a meeting and therefore enables a dynamic environment to grow. During the meetings, tacit knowledge was exchanged through social interaction which lead to constructive learning, group tacit knowledge, internalization, triggers, decision making and/or recall decisions. The decision-making process as well as recalling decisions are discussed in more detail in section three, the focus will lie on constructive learning, group tacit knowledge, internalization and triggers in the following section. These categories surfaced after the third round of analysis as well as through the literature.

As previously stated, the goal of the analysis relies on Nonaka (1998) and Takeuchi's SECI model and 'Ba', Ryan's theoretical model for the acquisition and sharing of tacit knowledge in teams and Clarke's Tacit Knowledge Spectrum. These models aid in the pursuit of understanding how tacit knowledge evolves from the individual to a group level as well as what sparks tacit knowledge to grow. In Table 5 these theories are compared to see their differences and likenesses.

'Ba' sets the scene for the dynamic knowledge exchange environment, where socialization, externalization and internalization can take place. In the three models used for analysis overlaps can be found. These were taken into account when analysing the data; however, the analysis that follows focuses on the following elements of each model:

- I. Social Interaction (Ryan, 2013):
Socialization (Nonaka, 2000), Knowledge in- and output (Clarke, 2010)
- II. Group Tacit Knowledge (Ryan, 2013)
Transactive Memory (Ryan, 2013)
- III. Internalization (Nonaka, 2000):
Tacit Knowledge Spectrum (Clarke, 2010)

IV. Triggers (Clarke, 2010)

Demonstrated below are the overlaps in the different models:

Table 5 - Comparison of the Theories of Nonaka, Ryan and Clarke

Nonaka	Ryan	Clarke
Socialization – tacit to tacit (face-to-face)	Tacit knowledge acquired and shared through social interaction.	Knowledge in- and output
Externalization – tacit to explicit (visual aids)	Tacit knowledge acquired by individuals through constructive learning.	Knowledge in- and output
Internalization – explicit to tacit (learning)	Individual knowledge / Enacted into transactive memory.	Process of acquiring and processing tacit knowledge (reflection – trigger – tacit and/or explicit element – existing knowledge)

The topics, in accordance with the theories found, narrowed down the search for tacit knowledge within the data. Social interaction is at the heart of the recorded meetings and is later categorized as socialization. The two main themes, individual and group tacit knowledge already stand out. The interplay between personal and group tacit knowledge will later play a vital role in the construction of the model as well as its contribution. In addition, constructive learning was also a focus of the data and its impact on tacit knowledge creation. Exchanging the expertise of each individual within the project group gave way to learning and accumulating knowledge. Then, Clarke's approach of a tacit knowledge trigger was added to the categorization. Tacit knowledge triggers are the most significant finding which will be further developed in the third research cycle. The final category

was decision making, which allowed the analysis of decisions made over time. This category is also further developed during the third research cycle.

4.2.3 Research Cycle 3

The third research cycle focused on pulling together the theories and data of the first two research cycles and putting them into context. During this phase, the recordings were listened to several times. First, the tacit knowledge triggers were searched for. This allowed the understanding of tacit knowledge surfacing. In addition, the material was analysed in accordance to the time of the meetings and the topics discussed. These discussed topics were then analysed for overlaps in subjects over time, meaning whether a subject or issue was discussed more than once. This helped build the decision-making process as well as looking at recall triggers in decision making. Reassessing and validating socialization, externalization, constructive learning and group tacit knowledge, combination and internalization was also central to the third research cycle. Colour coding helped categorize the extracts according to their tacit knowledge output (i.e. socialization, visual triggers etc.). In addition, these categories and their frequencies aided in building the model. The aim of the third research cycle was to answer the second, third and fourth research questions.

Social interactions are the largest part of a meeting, next to visual knowledge stimuli. Frequently, social interaction creates constructive learning by a team member asking questions. A specific topic leads to a discussion where questions are asked, answered and internalized by each individual in order to create new tacit knowledge. This can also lead to constructive learning, which is one of the most efficient ways to create and internalize new knowledge. When tacit knowledge is passed on from an expert to the group to clarify a subject the team's knowledge prospers and is enriched over time. "Individuals draw from the team tacit knowledge and create their own tacit knowledge. This is a background process which is dynamic and reciprocal relying on constructivist situated learning" (Ryan, 2013).

Figure 12 illustrates the amount of tacit knowledge found in each category. As previously stated, socialization was at the centre of the meetings and occurred 45 times. In addition, internalization and group tacit knowledge was also found 45 times. This demonstrates the importance of meetings and acquiring tacit knowledge in a group setting. There were 28 externalisation situations, 18 constructive learning and finally 9 combination. This information is the basis of the analysis of the data which is used in the construction of the model.

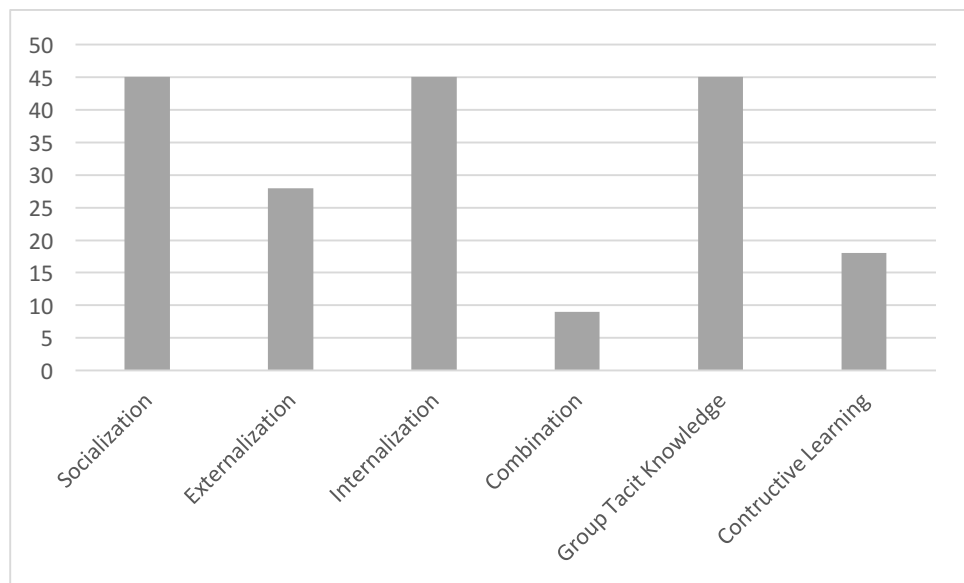


Figure 12 - Number of Tacit Knowledge found in each Category

This cycle also closely looked at the knowledge exchange of the individuals involved in the project regarding tacit knowledge triggers. The different tacit knowledge triggers emerging through the data will be shown and how they emerged through the data. Within each extract, triggers were found which allowed tacit knowledge to surface. Five main types of trigger were found during the research, some of them found in Clarke's (2010) model:

1. Visual Triggers:

Tacit knowledge surfacing through visual stimuli.

Looking at previous notes or looking at the software enabled tacit knowledge to surface.

2. Conversational Triggers:

Tacit knowledge surfaces through a conversation held within the team.

These are very frequent, here tacit knowledge surfaces while discussing topics related to the project.

3. Constructive Learning Triggers:

Tacit knowledge is enabled through a team member explaining and the others learning from them. The trigger is within the person learning from the explicit knowledge.

Constructive learning triggers surfaced regularly when the software development team explained the new software to others in the team.

4. Anticipation Triggers:

Tacit knowledge was exchanged by an individual in the group by waiting for the topic to come up or the meeting to take place.

5. Recall Triggers:

Tacit knowledge resurfaces through discussions or visual aids which seemed forgotten or not present by an individual.

The five main types of tacit knowledge triggers were found in several extracts. In total, conversational triggers were the most frequent, meaning that within a conversation newly gained knowledge allowed new knowledge to surface. Followed by constructive learning triggers 19 times, visual 18, recall triggers 7 times and anticipation 2. This is also shown in Figure 13. The trigger points are also used in the construction of the model for the construction of the model.

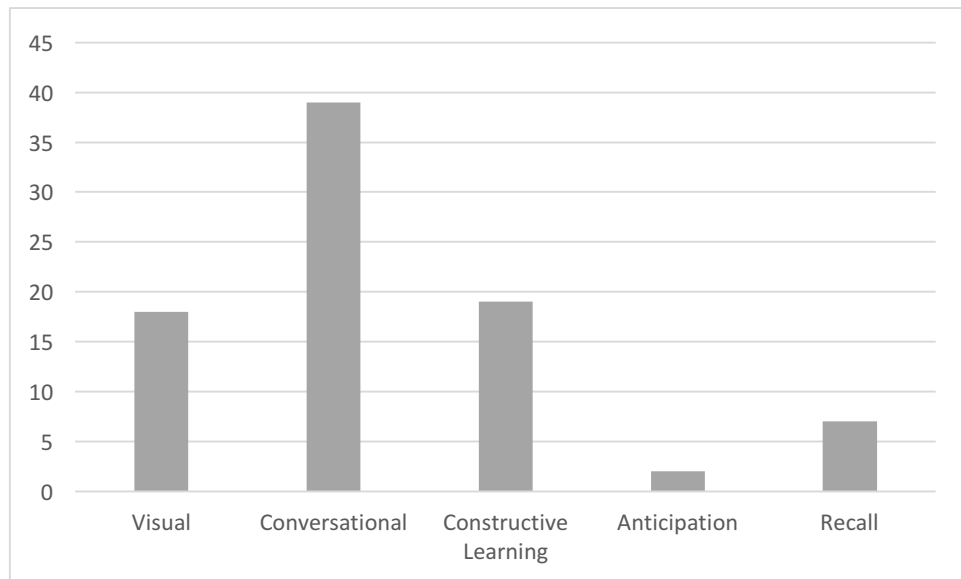


Figure 13 – Number of Tacit Knowledge Triggers found in Data

The third research cycle also allowed a comparison of the occurrence of tacit knowledge triggers and tacit knowledge. In Figure 14 the creation of knowledge and its relationship to trigger points is shown. It is evident that conversational triggers allow tacit knowledge to surface the most. Constructive learning as well as visual triggers are the second and third trigger which enables tacit knowledge exchange. Recall triggers and anticipation triggers were not found as frequently as the others and are therefore occurred the least. It can also be seen that combination and triggers are less likely to surface, whereas socialization, internalization and group tacit knowledge were the strongest tacit knowledge exchange factors. This graph helps to visualize which tacit knowledge exchange helps trigger tacit knowledge. The model will help to understand the trigger points and their importance to tacit knowledge exchange.

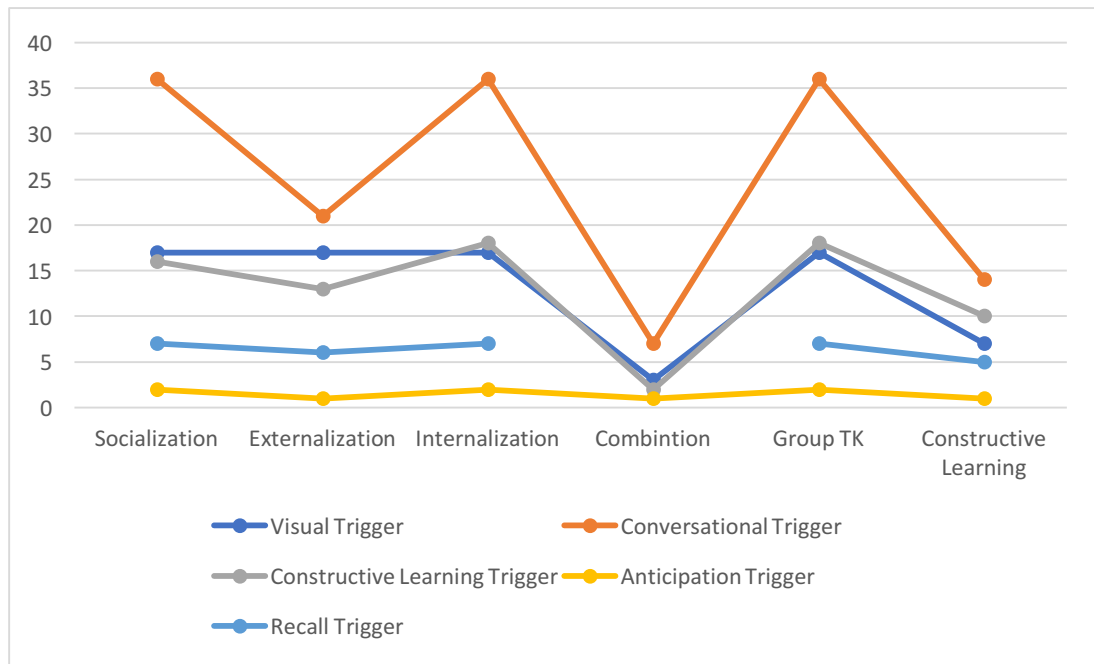


Figure 14 – Knowledge Creation and its Relationship to Trigger Points

Finally, decision making was analysed with the NDM model. The natural decision making model allows decision making to be analysed through yes or no questions and which actions were taken. Decisions made within the project team were assessed and four decisions could be recorded. Particularly interesting were the recall decisions, which were instances where tacit knowledge resurfaced through conversation. This tacit knowledge was linked to a decision previously made but forgotten. Through the assessment over time, four instances of already made decisions, which were not recalled, could be found. These were very interesting since it highlighted the need for meetings and social interaction within a project. Each project member benefits from the knowledge of their colleagues.

4.3 Summary

This chapter focused on the concepts and theories used to analyse the data collected during the meetings and findings. Having had a basis of analysis when starting the data collection process, the more refined theories used to analyse the data were researched in collaboration with the evaluated data. This allowed the data and theories to emerge over time and reveal the

tacit knowledge exchange and its triggers. The need to demonstrate the various ways tacit knowledge was evident throughout the project, the literature and results were shown to fully understand the evaluation process. The following chapter will focus on the evaluation process of the data and its research cycles. This will show each conversation extract and where the tacit knowledge categories were found within the data. In addition, the decision-making process will also be analysed.

Chapter 5: Findings & Analysis

5.1 Introduction

This chapter focuses on the evaluation process of the data. It is a summary of the research cycles where the results are discussed in detail. Throughout the chapter a number of indicative Tables are used to show the occurrence of tacit knowledge within the examples presented below. At the end of chapter subsections 5.2 and 5.3 a summary of all the Tables will be given. The SECI model, group tacit knowledge and constructive learning were themes which emerged during the first round of analysis, these are summed up in a Table in accordance to the tacit knowledge found after each extract. The seconded Table shown focuses on tacit knowledge triggers, which are the themes which emerged during the second cycle of analysis. The evaluation of the meetings gathered is the main focus of this chapter. The meetings are separated first by their subject matter, then by their time. Each extract highlights several parts of tacit knowledge exchange, which come through different channels.

This chapter aims to assess meetings held over a three-month period in the context of concepts and theories found in the literature. Tacit knowledge is highlighted by using extracts of meetings and linking them to relevant concepts. From the time when the data is collected, topics emerge again over time and are combined in a narrative form in order to understand the evolution of a topic. The analysis of the collected data was the greatest challenge during this research. Identifying tacit knowledge during a conversation and finding meaning as well as purpose was very difficult and as previously stated took several rounds of analysis to achieve.

In the following sections, research questions 2 and 3 will be assessed and answered through analysis of the collected data. To begin with, addressing research question 2 will demonstrate the exchange of tacit and explicit knowledge in a software development project. Knowledge input and output – socialization, externalization, constructive learning, social interactions – is the first focus of the collected data. This will be

complimented by a more detailed analysis of individual tacit knowledge and group tacit knowledge. Research question 3 will focus on decision making and will utilize parts of the data analysis of research question 2. Finally, a short summary will conclude the data analysis; results as well as research question 4 will be discussed in the following chapter.

The following sessions, seen in Table 6, were evaluated and recorded.

Table 6 - Recorded Meetings Table

Date dd.mm.yy	Duration hh:mm	Comments
04/03/13	00:09	Payroll and Sick Pay
04/03/13	01:38	Payroll and Sick Pay
04/03/13	00:01	Buro Licensing
04/03/13	00:07	Payments
05/03/13	03:08	Time, Payroll
07/03/13	01:19	Recruitment
07/03/14	01:58	Recruitment
11/03/13	03:34	Payroll, Employee List, 360,
12/03/13	03:58	Licensing, Payroll, Project Plan
25/03/13	02:11	Payroll, Pensions, Reports, Dashboard
26/03/13	01:06	Searches
28/03/13	00:59	Time
02/04/13	02:24	Payroll & Data Entry
04/04/13	00:30	Training, Employee List
04/04/13	01:26	Time, Job Application
08/04/13	02:16	Payroll, Project Plan, Training

The different teams involved in the project, cooperating with each other from each organization were analysed. The software development organization, the human resource organization and the customer. The interplay between the three companies and the knowledge passed from one organization to the next is essential in the understanding of the project and the knowledge passed on throughout it. In the centre of the project stands the human resource organization; they feed knowledge to the customer as well as the software development organization. They are crucial for the correct input to the software and for ensuring that all requirements are specified by the customer. Following on from this, the software development organization creates the platform for the software creation. The layout as well as the initial pages are provided by them. The HR consultancy refines the first layout from the software developers. Finally, the customer feeds the software developers as well as the human resource consultancy knowledge of the standard HR practices and needs of their organization.

5.2 Analysing the Data

5.2.1 Finance

Payroll is an essential part of the software for the employees to know how much they will be payed, when and which additions and deductions were made. Payroll hinges on working hours and is therefore connected to the time tool, the calendar. One of the great challenges of the payroll tool was the transition from the previous payroll system to the new one. The new software needs to be fed with current, future and historic data in order to calculate the right amounts. In the following section the discussion of payroll is assessed, in particular the interplay between team members, gaining knowledge from team members and using the new gained tacit knowledge to advance in the project.

Extract 1:

HR A: In an unrelated topic, we talked about sick pay, policies and rules last week. I do not have any up to date paper work from you guys. Could you send me the most recent copy?

CL A: I can send you the policies, because we did update it about 6 weeks ago, when we changed the sickness payroll for the organization. The long-term sickness absence. So, I can send that over to you. Could you copy in SD A as well? Thank you.

SD A: So, Payroll, while you mention that...

(Date 4.3.13)

During this conference call the HR consultant asks the customer for a document stating rules and regulations for sick pay. HR A has waited to ask CL A to send her this document. Through socialization the customer answers the question and agrees to send the document to the HR consultant, which is explicit knowledge. The update of the document was an act of combination where explicit knowledge is combined. The sick pay policies conversation triggers the discussion of sickness payroll, SD A has anticipated for this topic to come up, anticipation trigger. The dynamic environment allows a change of topic during the meeting. The main findings can be seen in Table 7 and 8 below.

Table 7 - Extract 1 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x	x	x	

Table 8 - Extract 1 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x		x	

Extract 2:

SD A: So, Payroll, while you mention that, we would really like to not switch off the old software and go live with the new payroll programming one hit, because from the HR side that's not an issue at all. But payroll is so much more important and our payroll is a bit different from the old software thing. So, we would like to put that off for a few weeks, just to make sure that the data matches.

CL A: That's ok.

HR A: SD A, can I ask a question about payroll. I completely understand what you are saying about the old software. Would it be best, bearing in mind that we are coming up to the end of the tax year to leave the old software running before we switch, or is that not going to make a difference?

SD A: I think it would be better to leave the old software running on the payroll side of things.

HR D: Correct, so we shut off at the end of march for that year end and then start the new payroll software beginning of April.

SD F: No, that's too soon!

SD A: No, because the go live isn't till the beginning of April anyway, so we won't have anything to match.

HR D: Right.

(Date 4.3.13)

Through the previous change in conversation, SD A explains why the launch of the payroll piece of the software needs to be put off. SD A has waited to talk about the extension of the payroll piece, recall trigger, this can be seen later in the conversation it is stated that the go live of the software should not be at the same time as launch of payroll. This triggers HR A through conversation to ask whether the old software should keep running, and whether the software should go live in April. The SD team, clearly states that this will not be a possibility, due to non-matching data. This extract

mainly focuses on socialization and team members internalizing the new gained knowledge of the project, this then allows a new common group tacit knowledge. Table 9 and 10 sum up these findings in their categories.

Table 9 - Extract 2 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x		x	

Table 10 - Extract 2 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x			

Extract 3:

SD A: What we are dealing with live, is that we can generate the payroll report out of the old payroll program and generate the payroll report out of the new software's payroll and confirm that everything is accurate, before we just switch off the old one.

HR D: So, what date were you thinking to make the final switch?

SD A: Well that's something to talk through with HR A and HR B, which we will hopefully get to this afternoon. Possibly it will be after the first monthly one, because we won't check the monthly one till that's done. We will go through this, this afternoon, whether after checking the weekly for a couple of weeks, will give us sufficient confidence.

The payroll is a big thing; I didn't realise it is a weekly payroll.

HR A: Even if it was a monthly, which I think goes out on the 25, they prepare it on like the 11.

SD A: If the old software would come out earlier, we could match it.

HR A: We have to keep in mind as long as they are running on the old software, they are paying double. I was surprised that CL A was a calm about it as he / she was. I thought he / she would ask about cost.

(Date 4.3.13)

This conversation, visualized in Table 11 and 12, takes place after the conference call with the customer, HR A talks about CL A not asking about cost. At the beginning SD A explains how they want to make sure that the payroll report is accurate once it is live, this allows the rest of the group to internalize the information through constructive learning. At the end of the conversation new group tacit knowledge has been reached on the topic of payroll and how the switch needs to be done in order to have accurate data.

Table 11 - Extract 3 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x		x	x

Table 12 - Extract 3 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x			

Extract 4:

SD A: So they might have a monthly London weighting allowance. What do you pay by period?

HR A: They have clothing allowance, first aid allowance.

SD A: So those sort of things. So it has a name, pay by period name, it has a pay type, it has a period it can fall into. It has to be authorized.

HR A: Every period?

SD A: Every payment has to be authorized. Sorry yes, it is authorized on their account and then its generated into weekly or monthly payroll as it gets signed off.

HR A: Would you only put in payments for that month or put in something...

SD A: ...you put it in as a go ahead, so when you set it up you select if it is set up for once or if it runs every month or.... For example, season tickets run over 10 or 12 months.

(Date 04.03.13)

In the example above the topic of discussion is Payroll – Additions and Deductions – Pay by Period. Table 13 and 14 display the extract's main benefit. The team is going through the software and analysing which information needs to be in- and excluded on the software pages. This is the process of externalization. SD A being the consultant for the software organization asks HR A, the human resource consultant, which information needs to be fed into the system – What the customer needs to pay by period. This conversation trigger allows a discussion of approval procedure, socialization, where SD A and B share their knowledge of human resources, organization procedures and software through constructive learning, while answering the question HR A recalls procedures of thing paid by period. These internal triggers allow tacit knowledge to surface. The other team members use the new knowledge and internalize it, hence there is a new level of group tacit knowledge. The knowledge exchange is dynamic where new gained tacit knowledge triggers questions and responses, tacit knowledge prospers within the group and the individuals.

(Date 4.3.13)

Table 13 - Extract 4 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	x

Table 14 - Extract 4 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
		x		x

Extract 5:

SD A: Is it a standard wage? You can have multiple standard wages such as London living wage. You can put pay on hold. So you know when the customer.... just going to get SD B up to speed.

[Explanation how the customer gets their employees together for a project.]

You can put a start and end date on hold. Pay date might not go through holidays.

HR A: So that is going to be the annual basic pay, sorry, the FTA isn't it? Oh no it's going to be FTM.

SD A: Yea.

HR A: Because over here you have the percentage haven't you. So will it work out?

SD A: I don't know, we need to ask SD B.

HR A: Because otherwise there is a lot of room for error.

SD A: The pro rata bit didn't work, the rest did. The standard hours need to be calculated to see hourly rate by default (on screen).

(Date: 4.3.13)

SD A explains the payments page in the software. During this process SD A stops to explain organization practice to SD B, who is not familiar with the customers practice. Externalization, the explanation of the wages, triggers socialization, face-to-face explanation of common practice, and then goes back to externalization, continuing the explanation of wages in more

detail, creating a new level of group tacit knowledge. Knowledge of what parts of the page are currently working and which are not allows HR A to internalize the knowledge and understand which parts of the software still need to be worked on. At times teaching can be interrupted during constructive learning due to group members focusing on previous or unrelated topics. In these cases, the topic is changed by a team member and therefore the internalization process of individuals is cut short. This extract is visually summarized in Table 15 and 16.

Table 15 - Extract 5 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	x

Table 16 - Extract 5 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
		x		x

Extract 6:

SD A: Multiple Pensions. Order of priority. So when they run out of money, this one comes first, this one comes next...

SD A: Say you are on 500 GBP a week and you get an attachment of earning because you failed to pay your child support. So the attachment will have top priority. There is a level in which deductions should stop.

HR A: Sorry can you just go back to the pensions type.

SD A: yea.

HR A: Just wanted to see where I can attach the file.

SD A: I think this needs a real thorough look, I am just skimming through it.

(Date 04.03.13)

The example above shows SD A, the software consultant, explaining the pensions page to HR A and then continues to the attachment of earnings page. The knowledge is internalized, however HR A still focuses on the past page, pensions, interrupting the flow of constructive learning and forces SD A to go back, before continuing on the attachment of earnings. Although the conversation has continued HR A recalls a previous page through a visual trigger. The conversation starts out with constructive learning, which then goes on to a socialized discussion, due to an external element, the software. SD A acknowledges that there is still work which needs to be done on the additions and deductions page, creating a group knowledge. This can also be seen in Table 18 and 17 where the main categories are demonstrated.

Table 17 - Extract 6 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	

Table 18 - Extract 6 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x				

Extract 7:

HR A: We would need to understand from your perspective (AC A) what we need with regards to the payroll export? Currently, when you receive the payroll data from the customer...you get the paper spread sheet. Does that get important or manually keyed in?

AC A: We manually key that in. It changes, it's not just Figures. It tells us what to do, we cannot ask the machine that. It is a very long spread

sheet. It tells us what the changes are and we manually have to change them. Presumably that is quite time consuming.

HR A: We like to work towards something which you do not have to key in, so it can automatically be imported. SD B is with me, whom is much more of an expert than I am. So what are your questions in regards to what they need?

SD B: I think the situation that we've got is that we've got an interface that we use for other payroll systems. Where it is all automatically inserted and what would be really helpful is, if we can keep the same format. It is quite similar, except that we have several spread sheets, rather than one. We have a spreadsheet with employee information such as their address and high level information, such as salaries or bank details.

Then we have another spread sheet with their additions and deductions. So you would have their staff number appear there again. So if they had several allowances, clothing etc., they would have 3 separate entries on the spread sheets. This gives the flexibility of adding as many additions as they have. If a new allowance is created that various people get, in your current spreadsheet you would have that in columns, now they are created by types. Then there is a separate spread sheet for addition of earnings and do you deal with pensions?

(Date 5/3/13)

Payroll is one of the largest topics during the project due to its complexity. How the data is currently transferred is discussed, which is combination, explicit to explicit. The explanation of common practice by AC A demonstrated constructive learning which results in new group knowledge, seen in Table 19. The work done however is a tacit act, since it is done individually by using tacit knowledge to prepare the explicit spreadsheets and feed them to the software. The social interaction above is socialization where through a call an act of combination is triggered. Through conversation triggers, tacit knowledge surfaces. AC A is asked to combine existing tacit

knowledge about the payroll with new acquired knowledge in order to construct spreadsheets. The other team members are able to internalize new gained tacit knowledge. The discussion continues where SD B explains the spreadsheets formatting and information needed by AC A to feed the software. This results into triggering a specific question regarding pensions, this can be seen in Table 20.

Table 19 - Extract 7 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x	x	x	x

Table 20 - Extract 7 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x			

Extract 8:

AC A: The pensions are the problem because obviously, we need to work out the formulas with the customer. It depends on the percentage; do they get 3%? There are other percentages depending on the salary, we take a lot of time working out what the pensions are due to the formulas. We need a specific Figure, so we know what to put in, so we don't have to calculate the formula. It is not a normal Pension, they calculate on salary sacrifice and all the employees which are not on standard salary rates.

SD B: So you need the percentage contributing from the employee and the percentage of salary sacrifice and from those you can calculate the actual percentage which they are contributing and the employer is contributing.

HR A: What I can do is show a spreadsheet which CL A sent to me a while ago to SD B so he/she can see what you are trying to describe because it is hideous. It is easier to understand when you see it. It's all to do that their salary sacrifice comes of their salary tax and pre lots of allowances. It is quite complex.

SD B: So if we come up with a sample spreadsheet to send you, with all the Figures in, which are relevant. You can approve it.

AC A: Yes, that would be best. This way we can say what else we need.

SD B: So to summarize, we have the employee spread sheet, with generic information and addition and deduction types. The employee addition and deduction sheet. Employee pension spread sheet and attachment of earnings.

AC A: AC B has asked CL A for a pension spreadsheet, since we always hit a wall when calculating the percentages of pensions. They want more information on their payslips for mortgages etc. So we are working on a new payslip with CL B.

(Date 5.3.13)

Socialization continues, discussions where tacit knowledge surfaces through team members, seen in Table 21. The topic at hand becomes more specific and issues with the spreadsheets become clearer. 'Ba' enables the team members to dynamically bounce knowledge and questions back and forth. Topics which are not yet fully understood are questioned between team members which are answered with current group knowledge. In this discussion it is shown that other team members, which are currently not present, as well as documents are needed to complete the spreadsheets and fill in the gaps of knowledge. This allows the individuals to obtain tacit knowledge, understanding which parts can be handled by present team members and which additional players are needed to complete the task at hand. Conversation triggers constructive learning, seen in Table 22, which allows a discussion of needs for the project.

Table 21 - Extract 8 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x		x	x

Table 22 - Extract 8 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x			

Extract 9:

SD B: Every month or year you get payed x amount.

HR B: SD A, for additions and deductions I put a red light, percentage or lump sum, I can't remember why I put that.

SD A: Was it additions and deductions or the fixed fee?

HR B: Fee.

SD A: We had several conversations about that. Can we show the numeration page? The first one was around fixed pay. I thought it was fixed pay.

SD B: There is pay by period. We had discussions if we need it. It should have shown what you get in a particular month. Pay period for a week. It would take your salary, additions and deductions, take all the bits and pieces and show a summarized set in this record. However, I'm not sure if it is necessary, since it's stupid. It just tells us; I think this is what we are going to pay you. But payroll is receiving a spreadsheet with pensions etc. separately. So it is just an extra nice bit, but we haven't decided whether we need it.

SD A: The bureau does not check the tax code number. We thought it would check it.

SD B: Who worked on this?

SD A: SD D.

SD B: I wasn't aware of that. So we should add a next box to not override the tax code numbers. I will just write that down.

HR A: What we really need is a detect and warn thing because sometimes especially with the ATI coming in. Well we should be getting the tax code and changing it from within and then inform payroll, but if they reconcile back to the revenue and tax check the payroll after it has been submitted, there are going to have to watch their data. Currently, the situation is only we change the tax code and inform payroll but ATI comes in, it could be changed from both ends.

SD B: Well, I don't think it's gonna be that the one that comes from the payroll should be put into a different field so you can see...that you can have a report where there are changes and then you've got a check. You can see what the old code was and you haven't lost your information.

HR B: Flag out immediately, it comes back in and it gives us something on the dashboard to say alert and go to that report and it will show us what it is that's changed and you can obviously then.

SD B: If we have a process where the received information from payroll tax code and then you'll report where that's different and it might be that then you choose to overwrite it manually or you check every one and choose whether it's correct.

(Date 5/3/13)

In this section of the meeting socialization and externalization are demonstrated. Visual, explicit elements, such as the red light on the page trigger a discussion within the team, hence the software services as a visual aid to encourage socialization, seen in Table 23. The page triggers a discussion of labour, since SD B, the head software engineer, was not aware who within her team was working on it. SD A, the consultant allocated the

work SD D, a software engineer. Project members are made aware of on-going work by looking at the additions and deduction page which triggers spiralling tacit knowledge and an equal understanding of the additions and deductions page, therefore they are part of Table 24.

Table 23 - Extract 9 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	

Table 24 Extract 9 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x			

Extract 10:

HR A: I tried to explain pensions, but I could only do it poorly. And I said that I only understand it when CL A explains it. So you could you please explain pensions to us, so then we are hopefully on the same page.

CL A: [Explains Pensions in detail.]

SD A: So it would just be a matter of searching on date of birth for the 1st of January, and maybe having a check if someone has an incompatible mark on their age or something?!

CL A: That's right.

SD A: Having the notes on the boundaries would be great, thanks.

CL A: [Continues to explain Pensions]

(Date 11/3/13)

Several days after the discussion about pensions, CL A is asked to clarify the knowledge gaps of the other team members. Through constructive

learning pensions are explained in order for the project to move forward. Explicit material is asked for the spreadsheets to be completed. The meeting on the 5.3.13 triggered the socialization, where internalization could take place and new group tacit knowledge was created. This is an example of expertise allocation and retrieval over time, where a specific project member within the group is needed to transfer their tacit knowledge in order to advance the group. This is represented in Table 25 and 26.

Table 25 - Extract 10 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x		x	x

Table 26 - Extract 10 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x	x		

Extract 11:

SD A: Do you see there, it says Payroll ID and when we met you said it wasn't payroll ID, you said it was the works reference.

HR B: We only ever had one ID in paper, so we used it for everything. We said ages ago that there is only one reference.

SD A: I thought recently you said there is a different payroll reference.

HR B: No we only have one ID number. I think AC B sometimes has a different number because they come back, that might have been it. So there are people that might have come back and he gives them a different reference number.

SD A: Thank you I... might have looked at the demo data and had a correlation with it. So that's fine.

(Date 11/3/13)

The extract commences with a visual trigger, where SD A refers a specific wording on the screen. This triggers a recall from HR B who explains the practice of this issue within the organization, constructive learning. The payroll ID and the misunderstanding might lead to problems with data fed into the system, conversational trigger where socialization leads to an assessment of another or larger problem. This is summarized in Table 27 and 28.

Table 27 - Extract 11 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	x

Table 28 - Extract 11 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	x			x

Extract 12:

HR B: You want to know what his salary is at the moment; you go into here.

SD C: We can rearrange that.

HR B: Also, we want that salary tab as a first tab, that's the first thing you put in. Then have additions and deductions next and then the rest kind of follow. In the order you do it sort of thing.

SD C: So, you probably want the annual, monthly and weekly basic pay in the beginning.

HR B: I think you should still see the dates.

SD C: The dates and the basic pay rates. Are those the most important ones?

HR B: I think so.

(Date 12.3.13)

The software pages are being altered in accordance to the wishes of the human resource consultants. The software serves as a visual trigger to start socialization and externalization. Table 29 and 30 shows these aspects. HR B internalizes the explicit knowledge and asks for an act of combination, which will be done at a later stage. Group tacit knowledge is created through SD C understand the needs of the HR consultant as well as the HR B seeing the software.

Table 29 - Extract 12 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x	x	x	

Table 30 -Extract 12 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x				

Extract 13:

CL A: I have a couple of questions regarding first the historical absence data. Can we put those in?

SD A: Yes, you can put in a historical, current or future absence.

CL A: That's really helpful. There is another thing, concerning what you just said SD A, we want to see an annual payroll review, probably through paper which means that effectively will be changing the pay during April. All this means really is that when we take the final data, we don't take it at a point where half the changes are done and the other half isn't. That would be a nightmare.

SD A: On the payroll, I assume you want the current numbers and the old ones. So effectively you would want last years and this year's payroll numbers?!

CL A: I think we would need both from an audit point of view.

HR B: What bothers me is that there is so much room for error. I feel like I should just be putting it straight into the data base, rather than trying to put it into a spreadsheet first, so I can look at it.

SD A: Why don't you do that?

HR B: At the moment I am doing additions and deductions and pensions, like I said before I'll give it to you but I need to go through CL C, I was basing it on the reports, but it doesn't match. AC B has some stuff that we don't have, we have some stuff on paper that he is not paying so is that because there is an end date that should have been put in or is it not relevant anymore?! Is it just a user error for not putting in an end date or is there stuff that should be paid but isn't being paid?! Do we need that to be resolved before the data goes in? I think it does, but how long is that going to take?!

SD A: I think we need to get data in there, it's easier to go through and say these 70 people don't have this or that, than to be waiting for perfection, when we have 200 people to check.

HR B: We have everything that is in paper but someone needs to go through and check what is right.

SD A: CL C?

HR B: She is the only one that can do it, but if we wait for him/her to do that, we will never get in, but if we load all of this in as being current, people will be getting things, that shouldn't get anything at all.

SD A: How about scheduling a call Thursday morning with CL C and go through it with her, so it doesn't go to the end of her pile.

HR B: You think she can answer that like that, or does she need to look at every PM?

HR A: I think she might have to look into them.

SD A: If we email it today, and schedule it on Thursday, she has some time to look over it.

HR B: What we are dealing with are records not being closed.

(Date 25.3.13)

Tables 31 and 32 show the summary of extract 13. The call on the 23.3.13 is a weekly meeting between the software organization, HR consultants and the customer. During the call pensions and payroll were once again the main subjects. At the beginning SD A asks for an evaluation of a spreadsheet with historical absence data, however CL A goes back to talk about payroll, changing the course of the discussion. CL A, the customer asks for an annual payroll sheet. HR B transfers her tacit knowledge by evaluating the situation of payroll and the spreadsheets needed to feed the system. He/she explains how the situation is currently dealt with and what problems might arise over time. This allows the group to understand the work of HR B, which creates a new state of group tacit knowledge. The spreadsheets which were discussed on the 5.3.13 and the 11.3.13 need to be constructed and fed the software. Although explicit material, such as sent emails or spreadsheets are talked about, during this part of the meeting no explicit material was exchanged. The catch up call, where key project members, specifically the customer, is put 'up to speed' services as a platform for group tacit knowledge to prosper. Work is allocated to project members and progress is reported. During the call socialization and internalization of knowledge is in focus.

Table 31 - Extract 13 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x		x	

Table 32 - Extract 13 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x	x		

Extract 14:

HR A: Do you remember CL B who is the management accountant, and he / she is going to take a more active role in payroll. He / she has asked for a tailored payroll report. And I asked what field she would like to have included? He / she has asked if someone else could also run the report and AC A is just part of the finance team. I think he / she is the project member who keys in all the overtime, so he / she is not a line manager, I don't think. So you couldn't tick to have it.

SD A: There is another role, called payroll admin.

HR A: That's what I see CL B needing, but I am not sure if AC A needs all of that, but I was just thinking, if CL B set up a saved search, maybe once a week, and emailed this report to himself / herself, could he / she just then wiz it to AC A?!

SD A: He / she could have it emailed directly to AC A.

(Date 26.3.13)

On the 4.3.13 the discussion about permissions concerning the account, CL B first started. CL A was asked what access the accountant needed to work. Weeks later this issue was once again picked up in a meeting, where a solution for the CL B was found. The conversation starts out with HR A briefing SD A of the previous events and then asking which permissions would be needed and what access and information can be sent through as well as granted in the software, a recall trigger, tacit knowledge previously acquired resurfaces. Socialization is the main course of the conversation, with short explanations of what is needed, constructive

learning. This allows internalization from both parties as well as group tacit knowledge and the conversation triggers allows tacit knowledge of the software to be transferred to other project members. A visual summary is seen in Table 33 and 34.

Table 33 - Extract 14 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x		x	

Table 34 - Extract 14 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x			

Extract 15:

SD B: Your additions and deductions are annually right?

HR B: The Figures are but they are paid monthly.

SD B: So the 230 pounds is actually spread over the year, no monthly.

HR B: When I am currently dividing the number by 12, in the old software it does not give out the same amount.

HR A: Should we ask CL A on Monday or Tuesday?

HR B: The Figured just don't make sense to us.

SD A: We need to compare the Figures, to make sure that they are the same. We will put in the information that you sent us and generate reports and send them out to AC A and D etc.

(Date 28.3.13)

This is an example of socialization (Table 35), where spreadsheets are once again discussed. SD B asks HR B for knowledge, where an internalization process takes place and group tacit knowledge is created. The conversation triggers (Table 36) allow tacit knowledge to surface and be exchange. Combination is talked about, where information in the spreadsheets are compared.

Table 35 - Extract 15 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x	x	x	

Table 36 - Extract 15 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x			

Extract 16:

HR A: We are importing the payroll data, and we have some discrepancies on the lists from AC B and ours. So we were wondering if you could make more sense of it. We tried to feather them out as much as we can, but there is still some information we are missing. Rather than pulling in wrong data, it's better for you to have a look over it. HR B has sent the spreadsheet to you that she made this morning.

CL A: The additions and deductions? Do you just want me to go through it?

HR B: So what I did with the reports...So it looks like there is some stuff that we shouldn't be paying and some that we should but are not. Some are missing on paper.

CL A: It depends on what report you ran; we have already identified that when you run a benefits report it doesn't take the end date into account.

HR B: I didn't run a benefits report. I literally put the data in paper into a spreadsheet, so if there is an end date I would have captured that. There are defiantly benefits on there that should have end dates that doesn't. I didn't want to make assumptions on data that I am not clear on.

[Explains missing parts on spreadsheet and what has been paid.]

(Date 2.4.13)

The meeting discusses the payroll spreadsheets which have been a topic over the past month. Through constructive learning socialization unfolds, and finally falls back into constructive learning. The process of acquiring tacit knowledge within the group and adding knowledge is an interplay between knowledge output, internalization and knowledge input, which allows group tacit knowledge to grow. Conversational and constructive learning triggers allow tacit knowledge to surface. In addition, HR B addresses the spreadsheets which were used to create the current spreadsheet, which was tacit knowledge combination. These elements of tacit knowledge can be seen in the Tables 37 and 38.

Table 37 - Extract 16 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x	x	x	x

Table 38 - Extract 16 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x	x		

In the previous section, the tacit knowledge flow over time has been highlighted relating to finance. Subjects were discussed over the course of a month, and reappeared in the meetings. Knowledge was built up and added throughout the different team members. Topics came up again and solutions to problems were found. The interplay between socialization and constructive learning gives way to seeing how tacit knowledge surfaces and what triggers it. A large part of the payroll problem is the comparison of spreadsheets. Later in the project sessions are held where only data is put into the system and the spreadsheets are completed. This action is tacit and solely relies on a person and a computer, so there is no verbal material to be analysed.

5.2.2 Human Resource Tools – 360-Degree Feedback

The human resource tool, the 360-degree feedback, is a way to monitor and assess employees in a matrix form. Employees are chosen to evaluate their line managers, line managers their executive manager and vice versa. The aim of a 360-degree feedback is to evaluate an employee from top to bottom and from bottom to top in order to gain a more balanced view. More precisely it aims to increase self-awareness, leverage strengths, uncover blind spots and develop skills. The customer chose this system however the HR consultants are in charge of developing the questions. From the software engineering standpoint, the tool needs to be programmed in a way that employees, line managers and executive managers are chosen and the results should not be shown to the participants, i.e. an employee creates a feedback for their line manager, the line manager should not be able to see the employee's feedback. In the following section the interaction between the HR consultants, software consultants and customer demonstrates the growth of tacit knowledge over the period of time.

Extract 17:

SD A: Now we are getting into linked records, we have done the core records. We talked about name changing, to be the item type.

Appraisal type. Standard appraisal, 360 appraisals and a scoring appraisal. So this is something to look at with SD B tomorrow.

HR A: My thoughts on the whole is that we will probably have to change some of that, but I am not quite sure to what yet, until we start building the form, and then work through every stage of the process. I think it will become clearer.

SD A: Is there something from the old software that could make it clearer?

HR A: No, because they currently don't use it. I've got draft one of the questionnaire done now, which I would be happy to send to you but it hasn't even been checked by CL A yet. While we're at it, you know we talked about the summary of the feedback and SD B asked what kind of format do you want it in? We just got some of the internet that CL A quite likes, do you want them now or should I give them to SD B?

SD A: SD B. The feedback is in the process engine, so that's his / her part.

(Date 3.4.13)

In the example above, externalization takes place by SD A showing the software to the other team members present, this is related to a visual trigger. It is the first time the HR consultants see the software, HR A comments on the first impression of the page, which results in socialization. The draft of the 360-feedback pages allows HR A to use tacit knowledge and relate it to the needs of the client. The discussion mainly focuses on formatting; how the information can be fed into the system and what information needs to be put in. Allocation of work and who needs which information is another aspect of the conversation. This is visually summarized in Table 39 and 40.

Table 39 - Extract 17 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	

Table 40 - Extract 17 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	x			

Extract 18:

HR A: The 360 stuff that you sent to me earlier, can we run through it now?

CL A: Sure. [Explains 360 questions.]

HR A: [Explains 360 questions, how they were made, categories and system.]

SD A: If you could give me those questions in a spreadsheet that is cluster – skill – question and a line, then I can pull them in. You can then pull them into question sets and set up some feedbacks. And then we can write responses, and give an example to CL A.

HR A: yes.

SD A: [Shows software – 360 pages.]

Will you use the same question set across multiple categories?

HR A: [Explains matrix of 360 questions.]

SD A: How many different question sets do you have in your excel spreadsheet?

HR A: There are four, what CL A calls categories of staff.

SD A: How many of those are going to get the same question set?

HR A: None...well there will be some duplicates cause for example the skills questions, are going to everyone, whereas the business leadership sets of questions are only going to 2, the senior managers. How does the system know what they are? (Employee, senior, line manager)

SD A: Cause you would do a search to find all the senior managers?!

HR A: It's not as easily categorized as that sadly. We know, because CL A gave us a list of each but...

SD A: So if we go back into the system, in employee record.

HR B: Is there an indicator that we could use on employee record? To categorize them, give them a reference number. Such as senior manager is category 1, line manager 2 etc.

SD A: It could be grade, I was thinking about taking it off that excel spreadsheet, and that putting each column as a grade 1,2,3 unto 8. So 1 was the top. Would grade do it?!

HR A: If we called it.

HR B: Is the feedback anonymous?

SD A: I thought it wasn't, but we know who we are sending it to anyways.

HR B: If it was, we could link a number to each person and then reference their position with it.

SD A: I think if it is anonymous it would go a bit under the fence.

HR A: That's what I think, but CL A isn't sure about that at the moment. I think she would like to have the option.

SD A: I mean you could take the name off; the thing doesn't go out anonymously because we know who we are sending it out to. So it's just a matter of how you present the results, by taking the names out.

HR A: I think if we could draft that both ways. So if we stagger over the year, I think we are going to start with SMG first. As a test group, it would be helpful to have a list, who are the reviewees for may or whatever it is.

SD A: [Shows how to send a feedback in software.]

(Date 11.3.13)

Constructive learning plays a vital role in this extract. The software is explained and information of how it will work in practice are discussed. The specialized knowledge of each team member is needed to complete this part

of the software. First, constructive learning through externalization sparks socialization, which bounces back and forth. During the discussion a larger, categorization of employee problem is discussed. How does the software distinguish between a line manager, employee or senior manager? Although the topic is the feedback, the employee centre, where all the employee information is listed needs to be evaluated. Previously learned information helps the HR consultants put together a whole view what is needed in the 360 feedback centre. The software developers at the same time try to find ways within the software to meet the needs of the tool. This triggers the discussion of whether the feedback will be done anonymously. Conversation triggers constructive learning and vice versa. This is also seen in Table 41 and 42.

Table 41 - Extract 18 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	x

Table 42 - Extract 18 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x	x	x	

The 360-feedback tool is quite complex from a software development point of view. Its linkage to each employee record and hierarchical structure of the software poses a visibility problem. Answers to the question should not be seen by managers or executives, but just by the HR team. Sending out questionnaires and what questions should be asked as well as who should they be sent to is the HR side of the problem. In this case the exchange of knowledge between the software developers and human resource consultants are essential in order create the tool. Finally, the missing knowledge needs to be extracted from the client.

5.2.3 Time (Working Patterns, Leave)

Another major piece of the project were the time pieces. The time unit of the software are all the pieces related to absences and working patterns. From a software engineering point of view time was very difficult to program due to its complexity. Within the system several aspects affect the calendar, such as sick days, working patterns, or hours worked. There is a large amount of data which time connects including payroll. In this section the software engineer plays a significant role due to the complexity of the programming. The HR consultants and the customer are made aware of features and limitations within the program as well as the customer explaining what is needed.

Extract 19:

SD B: There is still quite some stuff that is still in development for the moment, which has to do with extended time entry. can we leave that for the moment? So again this would be an alternative period time, I will change the name of that.

Does the salary record change?

HR A: I suppose it could, you could go on maternity leave.

SD A: You would need a log, where it says that a person is on sick pay or maternity leave.

SD B: Would that be a part of your process template? This week the salary drops to here etc.

HR A: Yea.

HR B: There needs to be a trigger between the employee centre and salary etc. Everything that goes through salary needs to show what it is.

SD B: So when you create a new record, you should have a maternity pay salary, or sick pay etc.

SD A: Is maternity leave just another pay type? It isn't passed through at all its calculated within the payroll program, because it's tied in with the recovery program, which is a certain percentage.

SD B: Which interface does it go on?

SD A: We don't have to interface yet.

SD B: No.

HR A: So what did the person say?

SD A: The people who work with statue pay, you just need to send over when it starts. But when it is contractual maternity pay then it would just be pay rate, I guess. Do we need a record for the maternity leave?

SD B: It should say medical evidence received, that's what you want. So you just need to change the wording.

Trying to think this through, when would we add a new record here?

SD A: When you know somebody is pregnant and then you just come back to top up information.

HR A: I am wondering if we need a date notified.

SD A: So we could put waiting on medical schedule.

SD B: But would you put that on a process template?

HR A; We only want them to add days into the calendar where they should have been actually working. So we can calculate the genuine days of holiday or leave. So if they are not due to work on a Monday, you don't want to count a leave on a Monday. So it will only be inserted according to their working pattern.

SD A: So the time sheet and calendar do the same thing?

SD B: Yes, you chose against the service item, if the item should go into the calendar, so what will happen it will insert everything into the time sheet but then it will pick and choose which ones go into the calendar

and which into the time sheet. So holidays will go into the calendar but not go into the time sheet.

SD A: You have a calendar in activities, which might show that a person is on holiday from x to y.

HR B: But you might not want someone to know they are on maternity leave.

SD A: But the time sheet is only working days, so you've got both options.

HR B: So whatever is going to show in the calendar is going to be seen by all employees, isn't it?

SD A: Yea, so you can see when someone is actually away, so you're not going to get them because they are on holiday or something.

HR B: So if a line manager needs to see that someone is going on maternity leave, but it hasn't been announced internally, that person is pregnant. So he / she can see it on his calendar, but not on the employee calendar?

SD A: You can make the employee calendar non-public. So people can't see it. Or you can only show it to specific people, but the calendar rules are sort of generic.

HR B: We need to be careful, because we don't want employees to see each other's calendars. Equally the line manager would want to see it.

SD A: You can edit your calendar, depending on the absence type.

SD B: We have two options for entering time, as an employee you can only enter yours, as a line manager you can enter yours and the time for your people.

(Date 5.3.13)

This phase in the meeting demonstrates how a topic spirals and builds itself up over the course of a discussion. To begin with, the software engineer acknowledges the work that needs to be done on the absence part. This then leads to an open discussion of what needs to be put into the page. Here, socialization leads to externalization, combination and internalization

and then spirals around. Once again, the looking at the software triggers the discussion of what needs to be changed or done in order to complete the pages, visual trigger. This then leads to conversational triggers, where questions and statements lead to either constructive learning or socialization. The software engineer's externalization of showing the program page to the team members leads to a conversation of what the expectancy of the page is. Combination takes place in the future, where the discussed name changes are put into effect. The name changes are important in order to fit the organization culture or HR culture are discussed. These parts of tacit knowledge trigger internalization processes which allow a new group tacit knowledge. The discussion spirals where maternity leave is connected to the payroll and which role it plays. Comparisons are made between maternity leave, a medical absence and holiday leave. Access from co-workers and managers to an individual's calendar need to be separated. During this conversation a clear view of human resource knowledge, working practice and knowledge about the software are combined. Tacit knowledge is exchanged from an individual to a team member, entering it to existing knowledge which then creates a group knowledge. This allows team members to respond and add to new gained tacit knowledge. Table 43 and 44 illustrate the different types of knowledge found in this extract.

Table 43 - Extract 19 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x	x	x	

Table 44 - Extract 19 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	x	x		

Extract 20:

HR A: This is where I get lost, the average day thing.

SD B: The new annual leave allowance form... [Explains process of average days.]

SD A: Could I ask for a field name change, from number of working day percentage to work days per week percentage.

SD B: Well, it might be that one week there are doing 1 day and one week they are doing 2 days, so you would then say it's 50%. But it wouldn't be, it would be on average.

SD A: So could we say average percentage work days per week?!
ok, if that makes any sense.

HR A: That's why I asked if it is annual leave or extended time, because for annual leave my view is if you are a day's PM, then the small issue should be to book a half day, obviously if you are an hour PM, you book hours. I think that's it for annual leave, but I can see where with other extended time you might need to look at hours. At the moment we have to book hourly absences as a half day, which is not ideal. So that is why I am asking, either or.

SD B: You are right it's just annual leave, in extended absence you can put how many hours you want.

HR A: Fine. Do you have some demo reports that you could give me to send out?

SD B: We are still working on them.

SD A: We'll put in the data that you sent us and generate reports and then we can give them to CL A or AC A.

HR B: I have some information that will transfer well, such as our payables.

(Date 28.3.13)

This conversation demonstrates a constructive learning situation where the software engineer explains the concept of average working days. HR A asks to further explain the topic, which was triggered by the pervious gained knowledge and the screen, externalization, showing the average working time, shown in Table 45. This results into a future act of combination where a name change of a field is demanded, which was cause by a visual trigger seen in Table 46. Following a conversational trigger s discussion of what is needed to generate the working patterns in the system and who needs the information. At the end, we can see that the group has a new level of group tacit knowledge to build on.

Table 45 - Extract 20 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x	x	x	

Table 46 - Extract 20 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	x			

Extract 21:

SD A: We now have the spread sheets; do you have those for me?

HR A: SD B did send me a list; I don't think I have worked on them yet. The extended time ones, I have to add to this list don't I?

SD A: yea. Do we know what has been done there?

HR A: The clients have all been done till December, luckily.

SD A: Is it sufficient to just have 2013 in there?

HR A: yea.

(Date 2.4.13)

This short discussion about extended time spreadsheets, which are needed to complete the time data, demonstrates the software division of labour within the group. Through socialization knowledge is passed from the software consultant to the human resource consultant. HR A needs confirmation of SD A whether SD B has really sent the samples to them in order to complete them, HR A recalls this through a trigger. They also discuss an act of combination where the data from 2013 is inserted into the spread sheet to feed the software. At the end of the conversation, through internalization, new group tacit knowledge is created. This is also demonstrated in Table 47 and 48.

Table 47 - Extract 21 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x	x	x	

Table 48 - Extract 22 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x			

Extract 22:

SD A: So these are working Pattern notes, standard hours a day, cycle hours or durations, so it works for 7 days and then you kick off the working pattern again. Does that make sense? For example, there are companies that do 9 days on - 9 days off. So their pattern lasts for 9 days, and then it starts again. Standard hours, some companies have them some don't. In NetSuite you can choose if you start day is a Monday or a Sunday. Whatever you set as the first of the week, so if you set Monday, it's going to be 1 and if its Sunday it would be 1. This is the working day. These are the standard hours, so if someone has a 6 hours, 4 hours, 6-hour pattern, then that would be in the next one.

HR B: So what number was that?

SD A: Uh, let me check, 8. So these are the templates, every template has an ID. We designate the Employee record to the ID of the template. So each employee has a template associated with them. This is the employee working pattern. So we have an external reference, which is the paper reference, the employee, then we have the post that she is associated with, because the working day pattern is related to a post. You might have someone with several post, i.e. a secretary and can also be a cleaner. She also needs non-working patterns. Is there anything we can help you with?

HR A: I already have the start dates and the end dates in, what I still need to do is the other bit. So I got the names...I need to do the work pattern days' bit. I'll do that tomorrow morning when I am here then.

SD A: I have to ask SD B for the excel formatting.

Then we get to time, I probably ought to leave that to SD B, because there is a lot of complexity around time. But essentially, we talked about changing the menus in the rest of the system and we need to pass the decision back through this record, because that is leave and absence. The sort of things you are looking at here are: absence, extended time, working patterns. etc.

(Date 3.4.13)

The topic is taken further over the past conversations; in the beginning the knowledge of the tool was transferred in the group in order to understand functionality. Now the conversation has become more concrete where the working patterns of employees are discussed and how they need to be imported into the system. Again, the more detailed view of the time unit needs to be explained by the software engineer due to its complexity and the tacit knowledge of the consultant is not sufficient to fully explain it, here visual triggers allow questions to surface seen in extract 50. The consultant explains through constructive learning the general outline of the page, but does not go into the functionality, the combination of visual and constructive learning triggers allows a more complete knowledge transfer. Once again, externalization, shown in extract 49, leads to socialization, where new group

and individual tacit knowledge is made. Allocation of missing tacit knowledge and selecting the team member whom can transfer is also made visible.

Table 49 - Extract 22 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	

Table 50 - Extract 22 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	x	x		

5.2.4 Recruitment

For the recruitment pages, the human recourse consultants sent their specialist to evaluate the pages. The current system in place for the recruitment is not centralized and many steps are not recorded in a system, but rather written down on a note. An important part of recruitment is to post available jobs online, and the plug-in for the major recruitment internet websites to the new software needs to be available. There was one very extensive daylong meeting between HR C, the recruitment specialist and the software developers. During this meeting, the software was demonstrated and needs of the recruiter were noted in order to match the recruitment procedure of HR C.

The following sections demonstrate the recruitment meeting on the 7th of April 2013, which is split into different sections.

Extract 23:

SD A: How do you know there is a vacancy?

HR C: So from beginning to end. Normally I would get a call from one of our clients or I possibly make a call, and they would tell me I have a vacancy. Then they send you a job description, which I then look at and tweak if I need to. Then I post it on a job website, then you receive applications, which currently come through to your email.

SD A: How many job sites do you post on?

HR C: Just one at the moment, but I think we would eventually like to have the capability on our own website. So I get the applications to my email, open them, look at them, make a decision, save the CVs to a doc file. At the moment I am trying to get together a type of candidate database, so I have got files for sales people London for example. So I could find ways to go back if I needed to. I haven't had to but if I wanted to.

SD A: Do you send a thanks or no thanks email?

HR C: Yes, I do, but once the AC Bas been appointed, I've worked on systems before that have a traffic light sort of system and you would view it, and you would think maybe, click orange and it would automatically send, you have been short listed kind of email to them and then you could go back and say the position has been filled. But I wouldn't do that till the end, because I would be scared that people would fall out of the interview process and they need more people. I call the people talk to them, go through the job description with them. Make sure they are compatible.

(Date 7.3.2013)

The conversation begins with questions which lead to constructive learning. The software consultant asks the recruitment specialist to explain the procedure of how a vacancy is uploaded online and how to deal with vacancies. The conversation, socialization, turns into constructive learning, externalization, where the team members can then internalize the information also seen in Table 51. It is a form of interrogation, where knowledge is extracted from one member, for the group. The step-by-step

walkthrough creates questions, which are then again explained in detail - constructive learning trigger seen in Table 52.

Table 51 - Extract 23 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x		x	

Table 52 - Extract 23 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x	x		

Extract 24:

SD A: So you put that into [recruitment website]., it then comes in as an email, so is there a connection from [the recruitment website] so it would come straight into your data base?

HR C: I'm not sure.

HR B: Yes, you can, I used to work with a system that could. You basically post the position through that site and it would feed through. In that system, as you said, it would traffic light the CVs. So you could reject, or put on hold or whatever. Oh yea [recruitment website].

SD A: So it's like a connector between the job website and your system

HR B: [recruitment website] is the best.

SD A: I will look into [the recruitment website] then. It would be easier to tie it in than for us digging into each site ourselves.

HR C: I've worked for a couple of agencies and they've all used [recruitment website]. The last place I've worked they used it there, but they didn't know the things they could do.

SD A: All of the websites kind of have the same details, name, address, skills that sort of thing?

HR C: I'm not sure, they must. You go onto [the recruitment website], and then you put in certain information, job, position, that sort of thing, and then you have [several recruitment websites] and then you choose the website.

(Date 7.3.13)

This extract demonstrates tacit knowledge which has been gained through experience. The software developer is trying to find out which data is extracted from recruitment websites and put into the human resource software. HR B steps in, after HR C is not sure whether data can be extracted into the database of the software. Through social interaction, constructive learning is created which aids in the further understanding of recruitment procedure and what data needs to be fed into the system. The conversation is triggered through constrictive learning; previously gained knowledge is transferred to the team members. HR B helps HR C when uncertain, recall triggers aid in the knowledge a more complete knowledge transfer. This can be seen in the Tables 53 and 54 below.

Table 53 - Extract 24 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	x

Table 54 - Extract 24 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x			x

Extract 25:

SD A: So you call the candidate, you take notes, you match it to the job description.

HR C: If I decide to put them forward, I would amend their CV a little bit. If it's not formatted the way I like it, I don't try to do too much because I want them to see what they can do.

SD A: So the CVs come in, and you have a word document...

HR C: Then I would take their personal details, and put a name on, because you don't want them (client) to contact them directly. Then I save them again, in the client folder. Then I email it to the client, with a cover mail.

SD A: Do you send them in a batch, or do you do it as you come in?

HR C: That depends, the last vacancy that I had I sent them in batches, but I had 300 applications in 3 days. I would send them in a batch, call them in one go, selected some and then I sent another batch of 5, but if it was a job where you only have 10 applications a week or hardly any, I would look at the CV call them and do them as they come.

(Date 7.3.13)

The variations in the recruitment procedure are discussed. Once again through social interaction, constructive learning is triggered, where the questions asked by the software development consultant mainly focus on formatting and quantity of CVs received and how they are then sent to the client. SD A internalizes this knowledge for the creation of the recruitment pages, combination. Table 55 and 56 show the tacit knowledge categories in this extract.

Table 55 - Extract 25 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	

Table 56 - Extract 25 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x			

Extract 26:

SD A: So do the job by job. So do you pay by applicant or by job? How do you bill the process?

HR C: The client gets charged per job.

SD A: So if it is 300, it is your misfortune.

HR C: Yea.

HR B: When someone applies you are only allowed to keep their CV for 1 year.

SD C: I thought 6 months.

HR B: We try to apply best practice, so if we thought someone had a good CV, we would contact them to keep their CV for other postings.

HR C: In previous companies I worked for you would have a client data base, and when a certain date would come, they would phone through and ask if they still needed work and then check their details and check that they are still looking and then keep them in the data base.

SD A: I wonder if it is another 6 months from that check.

HR C: Surely, if they consented you would have through so.

HR B: Yea.

SD A: Do you think there was an unsubscribe button at the bottom of a mass email?

HR B: I can't remember.

SD A: SD B and AD E will know.

HR B: I think it was defiantly 1 year, because I used to do the audits.

SD A: I suppose for employees it needs to be 7 years because of the work pension.

HR B: Yea. With their payroll we need to keep 6 years, but their actual files we were only allowed to keep for a year, but they weren't our employees.

(Date 7.3.13)

In this extract the topic rapidly changes, which sparks a discussion within the team. At the start, the billing process was the topic, which then led to a discussion how long an organization is allowed to keep a CV in their data base. This is crucial for the software, since an automated system can be put into place, where CVs are deleted after the legal amount of time in which they are allowed to be kept. Here the expertise of each team member are needed, since the software should be geared towards best practice. However, more people are needed in order to accurately respond to unanswered questions. This section demonstrated socialization, seen in Table 57, where tacit knowledge bounces back and forth between team members. This can then be internalized and is made into group tacit knowledge. Conversation triggers knowledge to surface and allows a free exchange within the group seen in Table 58. Expertise from each individual are used, the spiral of knowledge builds up.

Table 57 - Extract 26 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x		x	

Table 58 - Extract 26 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x			

Extract 27:

HR B: So what happens once you selected the candidates?

HR C: The client says we want to see xyz for an interview. Then I would go back to the candidate, see if they are available at that time, if they are, go back to the client and then confirm. I send an interview schedule to the client. Then I send a letter in form of the email, tell them where they need to go and attach the job description.

SD A: So how do you select a CV? Do you have a special set of questions or go from experience?

HR C: Experience, I go through the CV, see what they have been doing. Ask why did they leave this job, what was their salary? That type of thing.

SD A: Do you create questions for the interviews for the clients?

HR C: I do, when it's a new job they sometimes ask me. But that's really me searching on the internet trying to find out what the client needs to do.

SD A: Do you take care of references?

HR C: I will call them up, collect them and then give them to the client, if necessary.

So once, a PM is accepted, I will call them up, let them know. If they accept, I will inform the other candidates that the job has been taken.

(Date 7.3.13)

The recruitment procedure is further discussed, focusing on CV selection. The selection process is tacit; HR C refers to it as 'experience'. The CVs are picked apart and questions are asked such as, why a someone left a job. The pattern of social interaction leading to constructive learning during the meetings is emerging. Internalizing the knowledge provided the software consultant is trying to develop a complete software which allows a recruitment process to be lead step by step through the program. This enables a knowledge exchange that spirals and is build up over time. Conversation leads to constructive learning, which then triggers follow up

questions, socialization. Table 59 and 60 sum up the tacit knowledge categories in this extract.

Table 59 - Extract 27 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	

Table 60 - Extract 27 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x	x		

Extract 28:

SD A: At what point do you know the salaries and what you will be charging for it?

HR C: Normally, when they first tell me on the phone. What's the location, what's the salary, so you really know then.

SD A: Do they give a salary band or a fixed salary?

HR C: Some do bands yea.

SD A: For the moment the HR managers can see the recruit and the vacancies, adverts, costs etc. and build up reference data, sets of questions for different jobs for the interview panel or the referees.

HR B: Actually, some companies may do that themselves, they don't ask us to do it.

HR C: So is that what the client would see?

SD A: Well they wouldn't if it is in your database. But they could if you worked it into their data base.

HR B: Some clients will do it themselves and would need access there.

HR C: Say if I was doing recruitment for the HR consultancy, I don't necessarily want them to see that I had 300 applications. I don't want them to see the applications that I have got.

(Date 7.3.13)

After the first part of the meeting, where the recruitment specialist, HR C, explained her day to day business, how a vacancy is opened, the CVs come and are selected as well as the interview process and acceptance by the client, the software consultant shows the software to the recruitment specialist and HR B, the other HR consultant. Externalization triggers socialization, where limitations and access of the client to human resource data is discussed. This allows new group tacit knowledge to surface, and allows the SD A to tweak the software according to the human resource consultant's needs. Tables 61 and 62 sum up the extract and its tacit knowledge.

Table 61 - Extract 28 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	

Table 62 - Extract 28 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x			

Extract 29:

SD A: [Shows salary calculation and %.]

HR C: Could you not have like a drop down, 8%, 10%, a rate and then you can just type in the salary. What happens if I got a range? Can you only do that once I have a fixed salary?

SD A: What we could possibly do is put a tick that it is a range and then you choose an average. Once the salary has been fixed you take the tick out and put in the fixed salary. You could calculate the percentage by hand, it wouldn't be a big deal I think.

HR B: I think the calculation would be good, I have seen some invoices recently with minor calculation issues.

HR C: Eventually it won't just be me doing but other people too. Do I trust other people to then start doing their own calculations? I don't think that this should be manual.

SD A: So you would need, final salary checks on here.

HR C: Yes, especially when there is a band.

HR B: You need item, description, how much will they get paid, what's the rate, how many?

SD A: And if 3 come in at a different salary you need a separate line for that. So you also need a tick box that says it is banded. So, final salary to be confirmed. So this needs to go through an approval process, you leave it in, it's an internal document, so it's not going to the client, since you haven't finalized it. It gives you a notion of the cash flow.

HR B: That would be my guide, that's how much money is coming in. In most instances you would have to do a call, at the end of the vacancy or the job going through, being signed off. Then you can take the tick off and put in the actual salary. For her final invoice she will have to put that salary in. By the time it would get to me, I'm just sending it out.
[Explains post-recruitment process.]

HR C: So once I've created all of this, then it's a new sales order. I'm working on it, but then once it's been completed, I have my candidate, job that's filled.

SD A: HR B needs to know when it is ready for billing.

HR C: Exactly, so then do I have to go back into the order and amend it and send it.

SD A: You go back into the order, you put the final salary in, so the numbers are right. Take the tick out, mark it ready for billing, then either you send it to the client or HR B approves it and sends it to the client, whatever the internal process is.

(Date 7.3.13)

This part is still focusing on the new software, and what information has already been put into the recruitment pages. Once again, externalization enables socialization and constructive learning from the knowledge of each individual. Team tacit knowledge is created by a common understanding of human recourse practice and the engineered software. The software is a draft of the final version, and created by software engineers who still need expert knowledge to complete the pages. There is still the need of information such as salary ranges during recruitment and percentages paid by companies to the human recourse consultants. These processes are explained to the software consultant to be altered in the software. Constructive learning with visual triggers turns into socialization, which then again leads to constructive learning seen in the Tables 63 and 64. The cycle of effective knowledge exchange can be seen, since non-communicated tacit knowledge pops up by receiving and using new tacit knowledge, this allows a new common group tacit knowledge.

Table 63 - Extract 29 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	

Table 64 - Extract 29 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	x	x		

Extract 30:

HR B: What we haven't gone through here is the process steps. You can set up process steps where the process is the same thing that you do every time. So this needs to be attached, this has to be done in 2 weeks, it sets up a whole list of reminders. So you wouldn't miss it. So you don't need to keep a list to remind you what you need to be.

HR C: I'm just conscious that if I have to go back, and having to do things like that. Going back to the sales order every time, I don't...

SD A: It's not every time, you can say at the beginning, I think this is what we are going to bill and at the end say what you will actually bill and then you hit a button and it is sent to the client.

HR C: That's what I am saying, so say I had like 50 jobs on, and I got temp jobs and lots of stuff going on. Then I can like a new candidate to the job and then something pops up and says, is this the right salary?

SD A: You can automate it, but I would always want something to work manually before you work automate it. Otherwise, we could think we want it one way but then I think I want it differently, and the second thing is you want to know what it is doing and maybe it isn't quite right you know.

HR C: I don't mean that it automatically sends that off.

HR B: Maybe we need a final page where it shows the placed candidate etc. and then there might be a link or a text box that closes that job. Then it automatically brings up the next screen where you put in the salary and then it gets sent off for invoicing.

HR C: I'm quite good with things like that but the idea is that there are going to be more people working underneath me, and then I need to trust that they go back, change and then send it on.

SD A: And you don't want with a band salary, that you put in minimum and then it goes out with minimum.

HR C: How do I know when a job gets sent off?!

HR B: Can we have a box where HR C puts in the final salary and then have a pop up that asks, do you want to change the sales order and send out the invoice?

SD A: Possibly and the reason I hesitate is, when you have several jobs for a client, it could be that the number of things that need to be matched aren't worth the hassle, since they are linked to job reference numbers etc. What will be there is a link to the Dashboard where it shows unfinished sales orders, it's been past its end date etc. [Shows recruitment pages.]

[Shows how to link vacancies.] So the issue is, how we tie the sales order and the vacancy together? Manually is the easiest way because of the variations. Or it could be that we link it up automatically. The trouble with automating it, is that there are so many ways tying it in.

HR C: What I am conscious of is that there are a lot of layers to do a simple thing. Now I take a phone call, I write it down. I pull up a word document, type it in and it's gone. I am literally going to be spending x amount of time, putting in all of this information, and remembering to go there, instead of it just being one page.

SD A: How does it get billed at the moment?

HR C: I just do it in a word document. I just type in, what it is, what it costs, email.

SD A: Do you have any visibility from a higher managerial level on the works of when and what money you are expecting?

HR B: Not at the moment.

SD A: Do you want that?

HR B: I think they want that.

SD A: If they want that, it's going to add an additional complexity in. There is a lot of data here, and you don't need to fill it all in.

HR C: If we could have 1 page and then a tab down the bottom. I don't know I am just thinking of a way to incorporate it all into one thing.

SD A: A lot of this information is already in, so you just click between them.

I wonder if instead of creating a sales order, which is going to be created straight into an invoice. We create an estimate and then the invoice may or may not come from the estimate.

HR B: Yes.

HR C: Yes. You could have something once you have completed the estimation, you click on save it brings up an option, would you like to create an estimate for this vacancy?

HR B: You might have jobs, which might be coming up but you don't know when. You don't want them in the system but you want them ready, because you know that PM is going to put that job on in 3 months. You have already done the ground work and then you can put it on live.

(Date 7.3.13)

This long extract is an evaluation of current software from the recruitment specialist, and it demonstrates the knowledge spiral between the software consultant and the human recourse specialists. HR B, the human resource consultant focusing on accounting helps navigate between the software developers and her colleague. The first impression of HR C is that the procedure is quite complex in comparison to what they are doing at the moment. A fair amount of her work is not visible or traceable to others in the organization and once a bigger team is in place the visibility will play a more vital role. Many of the steps are done manually and not electronic and are therefore not stored. Assessing the work done by employees makes this process more difficult. The previous gained software knowledge from HR B combined with her human resource knowledge allows a more complete opinion towards HR C's doubts. The screen commences a discussion, externalization to socialization, this allows individuals to share their knowledge through constructive learning, internalizing it and allowing a new

team tacit knowledge to surface. Once again, the combination of visual triggers and constructive learning allows socialization. This is a vital process for knowledge to be exchanged and spiral in 'Ba'. The categories of tacit knowledge are summed up in Tables 65 and 66.

Table 65 - Extract 30 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	

Table 66 - Extract 30 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	x	x		

Extract 31:

SD A: [Explains current system.]

HR B: So when HR C is working on the HR client, how does he / she know when she is working on a specific client?

SD A: That's a good point, we need a client field up here.

HR B: She is going to need a client area where there a general notes and conversations, going on about vacancies and follow ups and stuff like that.

HR C: So I log on and go on a client and call them up and update my clients.

HR B: For now, it's not clear where we are.

SD A: Can we work on this once the customer goes live, since it is quite a lot to do?

HR C: If CL A needs 20 people in 2 weeks' time for 2-4 weeks, does CL A do a search for bank people that have the card (CSCS) and then send a mass email or create a vacancy? She doesn't want to advertise for it?!

SD A: She can do an internal advert, so the people at the customer can search for it.

HR B: But these people don't work for the customer.

SD A: So a quick email. I don't think it's a vacancy necessarily. It could just be a group email.

HR C: You could search for Bank people that aren't working for the customer at the moment? So external with a CSCS card and then she could send out a mass email.

HR B: But then wouldn't you want it to be a vacancy? Like that when the responses come back in, its already in the vacancy. And then you can just go yes / no and then appoint the people.

HR C: So how do you send out the email?

SD A: There is a mass email feature.

HR C: So he / she goes into the vacancy, he / she creates it, 20 bank staff start on the 18th, 2-4 weeks and then she would do a search for the candidates?!

SD A: The way it works is... [Explains internal vacancy process.]

(Date 7.3.13)

Parts of the system are still unclear or not finished at this stage. The system is aimed to be used for several clients of the HR organization; therefore, when the recruiter works with the system, it has to be clear whom to post a vacancy for. Constructive learning from SD A triggers a conversation between the HR consultants and the software consultant with the help of visual triggers. The conversation becomes more precise over time, where the needs of all clients of the HR consultancy become important, unlike before where the client to whom the system is tailored for was always

the main focus. The usage, how the system is used when a recruit is needed is discussed. Here externalization leads to socialization and then creates new group tacit knowledge. Conversational as well as constructive learning triggers allow externalization and socialization to take place. This then enables a dynamic environment where knowledge spirals between the team members and builds up accordingly. This is shown in Tables 67 and 68.

Table 67 - Extract 31 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	

Table 68 - Extract 31 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	x	x		

Extract 32:

HR A: Presumably appointed is the last stage.

SD C: Would you like to have in arranged how you use it, what comes first and last?

HR A: Yea, that would be nice.

SD C: Or alphabetically?

HR A: Progressively would make since, wouldn't it? But that wasn't actually my queries. Say there are at the offer stage, if you tick appointed employee, should it then change to appointed? I am asking if it should, not it should.

SD C: You can appoint the person, but still be waiting for the signed contract. Or he is an appointed bank staff employee.

HR A: True, so they don't need to be necessarily linked.

(Date 2.4.13)

The short example demonstrates the discussion to find a solution to an arrangement problem within the system. A summary of the extracts categories is found in Tables 69 and 70. Here self-reflection of HR A can be seen. Through socialization the discussion of protocol arises. Here, HR A explains her inner thought process what needs to be done and asks SD C to clarify this thought. Conversation triggers lead to a thought process, which allows knowledge to prosper.

(Date 7.3.13)

Table 69 - Extract 32 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x		x	

Table 70 - Extract 32 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	x			

Extract 33:

SD A: The option would be verification type, references filed out, education level check or you just do a search for the references. You can do it either way, depending on what's easiest. It might be that verification is easiest.

HR A: Both sound logical.

SD A: If you do references in here as well as having references, you might find people won't want to fill in references.

HR A: Ok, let's leave references where they are meant to be. So what about the education check, where would we put that?

SD A: Well you have verifications in there, so if we look at an employee and look where we might put it. We might need to put some more tick boxes.

(Date 2.4.13)

This discussion mainly focuses on where to put information within the recruitment system. SD A explains the page, through visual triggers socialization begins and options where to put which information is launched. The order in which things are processed is important for the configuration of the page. Tacit knowledge from HR A is transferred and utilized to achieve a logical order of the recruitment page procedure. Tables 71 and 72 illustrate the tacit knowledge categories.

Table 71 - Extract 33 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	x

Table 72 - Extract 33 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x				

Extract 34:

SD A: Using bank staff to fill a post for a new project, would they be job applicants, so raise a vacancy record and then look at who you are assessing against the vacancy or is there a smaller record where there is a connection from a bank staff to a vacancy, groups number of people. Is it enough just to say these have been invited, these have responded, these are the ones we have appointed? Or would it be a full blown recruitment type?

CL C: The 2. one, the smaller scale.

SD A: [Explains process.]

HR A: SD A, presumably there would then be a trail of the amount of times you approached a person, so you start to build a picture of how available people are.

SD A: [Explains where to find internal CV.]

SD A: Yes.

CL A: [Explains bank staff categories.]

(Date 11.3.13)

The client has a need for staff on a project to project basis. In order to find staff which are qualified to do the job as well as whether they are available, the HR tool needs to be able to assess and filter out staff. SD A asks how the HR organization handles these types of vacancies in order to tailor the software to their needs. Socialization leads to constructive learning through a conversational trigger from SD A, seen in Table 73. Learning then triggers a question, through the internalized knowledge, and falls back into constructive learning, visualized in Table 74.

Table 73 - Extract 34 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	x

Table 74 - Extract 34 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x	x		

Extract 35:

SD A: Should you be able to put references into an employee record or job record?

HR A: yea, I think we need to be able to do it either way. Either they followed the entire recruitment process and they followed through or we have this new person to start on Monday and put him straight in.

(Date 25.3.13)

This was a short query from SD A to make sure whether recruitment data should be linked to an employee record. The employee record is produced once an applicant has been selected for a job. The topics overlap and knowledge needs to be linked to different parts of the system.

Socialization triggers a short response to resolve the problem and allows SD A to internalize knowledge. The Tables 75 and 76 show the tacit knowledge found in the extract.

Table 75 - Extract 35 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x		x	

Table 76 - Extract 35 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	X			

5.2.5 Employee Centre

The employee centre is the heart of the software. Within the employee centre all information related to the employee is stored and then linked to the other parts of the software. Filling out the employee form is the last step of the vacancy filling process, where a candidate becomes an employee for the organization. Information such as recruitment information, salary, working patterns and disciplinary and grievances are linked to the profile of each person. Each employee will have a limited amount of access to their profile, where they can send a change request to the human recourse team for a change in address or similar information. In the section below knowledge from the customer, human resource consultants and the software developers come together in order to build the information stored within the employee centre.

Extract 36:

HR B: What access does CL A want for CL B?

HR A: CL B, who've been come aware of over the past couple of weeks, who I think is relatively new?

CL D: Yes, he/she's the replacement of the management accountant in the finance team. (..) She has been delegated to payroll responsibility.

HR B: Once the software is set up, what kind of access do you in vision CL B having?

CL D: Well, I don't want him / her to have any change access. It is useful for him / her to be able to access records related to pay and/or benefits.

HR B: There is a role on the software which is called payroll manager or payroll administrator, which in our working we haven't got there yet but I thought it would be helpful to know which kind of level of access you would want him / her to have so I can bear that in mind once we start looking at that access role.

HR A: That sounds good.

(Date 4.3.13)

Permission and access to different records within the software are discussed. HR A asks CL A through socialization which access CL B is allowed to have. CL A then internalizes the information and responds creating new group knowledge, the conversation triggered this response. At the same time, HR A gives a status update on the permissions, allowing CL A, the customer, to know the status of the project. 'Ba' enables the exchange of tacit knowledge and its build-up over time. This is also shown in Table 77 and 78.

Table 77 - Extract 36 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x		x	

Table 78 - Extract 36 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
		x		

Extract 37:

HR A: So we are looking at it from the employee perspective?!

SD B: yea

HR A: You know how people that work for the client and live further away from the working site, would have a home address and a temporary address when they are working? How are we going to put that in?

HR B: I think we wanted to put a tick or something, to show where they live.

SD B: The issue we've hit, if we let them access NetSuite so they can change their contact details, then they get access to the emergency contact details, which is the really basic contact details. Either they

can see both or switch it off entirely. So for now we have switched it off. So they can only see the main address, but they can supply another address for you.

HR A: We have to give thought on which address we let them see, which presumably would be their current one. So we leave it as it is.

SD B: We can have a change request form.

HR B: Can we have a notes field, so we can put in they will only stay for 6 months.

SD B: Good idea. Would that be a field you would want them to see?

HR A: Yes, that would be a good idea.

SD B: [Explains categories.]

SD A: Just something that CL A raised yesterday, is that they have a very rare occurrence of agency workers. So I just wanted to show you the list, where you see the range of options you've got. There are three sorts, employees, apprentices and shareholding directors. So I wondered if it is at the right place. Is it used as an employee search at all?

SD B: I think it is in one search but not generally. It can vary, they can start as an apprentice, become an employee and then a share holding director.

SD A: I think it was the Employee Record not the Employee Centre, that we will put in the last 3 tabs.

HR A: Yea, that's what I've said.

SD A: Sorry, I just went back.

HR A: Oh no no, we started at the one end and then we went to the other end. It was the 3 in the middle.

SD B: Should they see their tax code?

HR A: I think they should see their tax code and national insurance. When you get a new tax code notice, you should give it to your employer.

SD B: And you NI category?

HR A: I don't think people know what that means, so no.

SD B: What about checks, verifications? We could put in check the visa but we can also put that elsewhere. I hope it is not confusing that it covers all verifications.

HR A: I think verifications is the best option.

SD B: The question of changing position to post. If we've called position, it will link to the organizational post and then we've got post start and end date. So I am wondering what is the difference between a position and a post?

HR A: We discussed the topic with SD A, and she explained the process, and the difference between the two and I said we need to name them different.

SD B: The problem is that here you link them together where they are in fact the same.

HR B: I thought that it wouldn't be the post that ended but it would be the position for that person that ended but the post is still there. So for the organization the post is still there.

SD B: So we need to change that to position start and end date.

HR B: The employee doesn't need to see the post information.

SD B: Let's just hide that.

HR B: Where is the post information held?

SD B: In the post records, under organizational cases.

You can create process for customers, sales leads, employees.

Is there anything you have to do within a week or month of them (employees) arriving?

HR A: Contract has to be issued within 8 weeks.

SD B: I presume there is stuff you have to do before they arrive?!

HR A: Yea.

SD B: So the first thing you have to do is process 1., then the key date name/start date. So then you set up stages.

HR A: Send a letter to the employee. Send acceptance

SD B: Should we say that these people already agreed?

HR A: Send CRB form

SD B: prepare desk.

HR A: notify IT

SD B: Let's put in mid-way review, which is the next one in the sequence.

HR A: What would be great is when you set up the real client engine, you could give us access so we can start building the processes.

SD B: Once the steps are set up, we go to an employee.

HR B: Can you start up a process automatically? Like when I put in a start date, will it automatically launch the process?

SD B: You'll see the list of things you have to do. Suppose we had created a disciplinary template, we clicked on create process steps. Then it inserts all of these fields for you.

HR B: Oh no, I mean when I start a process will the process template be attached automatically?

SD B: We can't link it. [Shows process.]

It's probably best that you try it out yourselves.

HR A: Yea, it's one of those things where I feel like I want to try it out.

I have a question; it has just fallen of my head. SD A told us yesterday that NetSuite have provisionally agreed on the Buro setup. That they still have to sort out some details but it looks promising. So say we

had that setup for us, the human resource organization, and for each of our clients. How would the process steps work then? Could we set them up at both levels? Like generic ones for each of our clients and general ones as well?

SD B: What we are building is an import/export process so you can transfer the exported data from one account to the next.

HR A: So if we do best practice templates we could export or import them to each client.

SD B: Yes.

SD A: Have you looked at the process engine?

HR A: It looked very straight forward and made a lot of sense, but as soon as you're in the real client NetSuite, we can have a login and put real processes in.

SD A: That's interesting.

(Date 5.3.13)

In the meeting, SD A, SD B, HR A and HR B are present, which enables a fruitful knowledge exchange. The employee centre from the employee perspective is discussed and assessed which information needs to be displayed. The interplay between SD A and SD B enriches the knowledge flow, here software engineering knowledge and software consultant knowledge come together and makes a more useable knowledge flow for the HR consultants to take in. At the same time the HR consultants can accumulate their knowledge and pass it on to the software development team. SD B explains the problems they have encountered with the programming and that administrative decisions need to be taken in order to solve the problem. This allows socialization to take place between HR A and HR B to decide what is needed as well as SD A to give feedback. SD B supports SD A in the process engine part of the software, since it is quite complex. SD B shares the tacit knowledge of the software acquired during the course of the programming with HR A and HR B allowing them to

understand the range of the software with its limitations and opportunities. This allows a more complete view of the project and group tacit knowledge to be reached. The interplay between externalization, socialization and constructive learning allows internalization and group tacit knowledge to prosper seen in Table 79. Visual triggers, such as the software, allow an internal process to surface socialization and constructive learning. This then allows conversational triggers and constructive learning triggers to take place and launch the spiral of knowledge shown in Table 80. Remembering previously gained knowledge allows recall triggers to launch socialization and to resolve problems and issues.

Table 79 - Extract 37 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	X

Table 80 - Extract 37 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	x	x		x

Extract 38:

SD A: [Shows Employee Centre.]

HR B: That's weird, I was doing this post. I was in the middle of the record and I put in the information and then I pressed save and then it said I need a salary. Then a pop up come and asked if this is his substantive post and then it took me back to the employee screen, it didn't let me put a salary in. It ok me away from where I was.

SD A: Either, we haven't got a working pattern, and that is tripping us over or we have 2 things going on and there are tripping us over. I just went back and there are 2 things that SD D is working on related to posts.

HR A: We need to talk about the wording.

SD A: What would you like?

HR A: The data we have given to SD B, some is on a need to know basis, so just the E and D person and then there will be some which the line manager needs to know.

SD A: Equality and Diversity is going to be confidential information and the other is going to go into the verifications.

HR A: Is a chronic asthmatic, for disability and discrimination purposes that would be considered a disability because it is a lifelong problem. So it's the kind of thing I would tick that new disabled box for and it would also fall into medical alerts because the line manager would need to know. Or, could we add it to medical alerts? Does this medical alert have to do with disability, or something like that?

SD A: And then have a housekeeping check. But if they are not registered in E and D, does it count in E and D?

HR A: If it is something that affects you doing day to day activities it is a basic disability. For example, if you have cancer are you registered disabled?!

SD A: No.

HR A: Cancer and HIV are the 2 named exceptions under DDA, as soon as you are diagnosed with either of them, you are automatically covered by E and D.

(Date 11.3.13)

At the beginning of the conversation SD A explains the employee centre, while listening, HR A is using the software. This text is ended due software engineering work from to SD D. This was a constructive learning module where HR A was able to use the software and have teaching assistance from SD A. They then move forward to information an employee can see, here the importance of HR knowledge from HR A is clearly reflected in this extract. SD A understands, through constructive learning what is

needed in the software in relation to equality and diversity and what the parameters for E & D are. Externalization and constructive learning lead to socialization, which then falls back into constructive learning. This allows group tacit knowledge to prosper from the software engineering as well as the human resource side. Visual triggers allow discussions to surface, which then can lead to conversational triggers. Tables 81 and 82 show a visual summary of the extract.

Table 81 - Extract 38 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	x

Table 82 - Extract 38 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	x			

Extract 39:

SD A: [Explains searches and how to create lists].

HR A: There is one-line manager on the client's side who currently can see that data, can we do a safe search that just he / she can see, or would that breach his / her access level?

SD A: He / She could run the search, but he / she could only see his / her downline because they report to him or her.

HR A: That would be perfect. All inter project managers report to him / her, so that's perfect.

SD A: I'll make a note of that.

(Date 26.3.13)

Table 83 in indicates that through constructive learning and visual triggers HR A remembers a manager from the client who is able to see a specific set of data. The recall trigger enables socialization to take place seen in Table 84. Conversational triggers, then allow SD A to solve the problem, where the line manager is able to see all the relevant information he or she needs. This allows HR A to gain new tacit knowledge and SD A more fully understands the structures needed for the client, creating a new common group tacit knowledge.

Table 83 - Extract 39 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	x

Table 84 - Extract 39 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	X			x

Extract 40:

SD A: We have signed off all the employee stuff, we still have some changes and amendments that we have to make, but we are making very good progress and we have moved into some of the subsidiary records around post and status and different bits. So, it's going well. One is a little bit cautious with projects.

(Since it is buro license) the only option will be for people to set up as a line manager, or employee. The HR managers will be set up externally, so it will be designated, because they use full user licensing.

HR A: Interesting, so does that mean (uhm), so say if the HR consultant sets up as a buro and we would use 'chamber' as our client, does that mean no one at chamber could be an HR user?

SD A: No, you'd have to request that they set up as an HR manager. You know, I don't know we are still figuring it out. The reason is the HR manager is a full user, and NetSuite want them to buy all of NetSuite. So for the moment it looks like an HR manager can conFigure as many Line and Employee Centres as they want, that they licensed for but to set up someone as an HR manager would be an admin task.

HR A: So someone who has a certain level could be an HR manger when purchasing the whole license. Usually you only need one person per Organization who needs this kind of level, so that would be ok if it were you (HR B) for example.

(Date 3.4.13)

Licensing and payment of the program is a main focus of the project. NetSuite being an ERP which provides solutions in many fields, such as finance or CRM, is quite costly, when only using it as an HR tool for employees to set up their sick days and check their payments. Therefore, only HR managers need the whole license since they use the software for more complex tasks. The conversation starts out with constructive learning which then triggers an internal thought process in HR A resulting in a question seen in Table 40. SD A replies at first but then in the internalization process, through an internal trigger, acknowledges uncertainty and is not able to answer the question at this time. At the end of the conversation new group tacit knowledge is created, even though not all questions have been answered shown in Table 86.

Table 85 - Extract 40 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x		x	x

Table 86 - Extract 40 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x	x		

Extract 41:

SD A: I've got a question for SD B, which was where does the - I found out the answer in the end actually - where is the applicant status. So you can search everybody who is bank approved or limited bank.

HR A: I think realistically I don't think they will get back film on this data. It will be one of those things, as we move forward we will start putting them in. Or if you run a search and you find a person, you will add their form and data.

SD A: Which file would have employees for which working pattern?

HR A: Oh, that was weeks ago, uhm it would have been called something like – is there one called employee working pattern or...

SD A: Yea there is.

HR A: That's the demo one. I am sure I made one...

SD A: Yea that's another thing, NetSuite is doing an update on our system tonight, so somethings like the L&D external provider the ways that's managed will be affected by that, so I didn't put that in yesterday, know that...

(Date 4.4.13)

In this extract SD A needs to get knowledge from SD B. Internal triggers allowed, through reflection, enabled SD A to find a solution herself. However, HR A has a more progressive solution to the problem, and therefore SD B's knowledge input is not needed. Recall triggers allow HR A to remember a working patter created an earlier stage. An update from the NetSuite platform stalls some of the upload progress. This accumulated

knowledge from the different team members allows a new common group tacit knowledge. The main features of the extracts are summarized in Tables 87 and 88.

Table 87 - Extract 41 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x		x	

Table 88 - Extract 41 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x			x

5.2.6 Disciplinary and Grievances

The disciplinary and grievances (D&G) pages, allows HR employees to track information on each employee. The documentation and confidentiality of this data is central due to legal issues which can arise.

Extract 42:

SD A: They are pretty similar as in the appeal process is similar in each incident, it is linked. An appeal is linked to a disciplinary.

Would a capability ever be raised externally?

HR A: I suppose it could be a customer complaint or something?! But then it would be dealt with by HR, so you could simply add the information. It would then just be a complaint, which is then assessed whether it needs to become a disciplinary action

SD A: [Explains allegations, responses and investigations].

HR A: What is the difference between, investigation by and chairman?

SD A: If it is a serious thing you have a chairman and an investigator.

[Explains penalties, notes and dates].

HR B: Should we have the notes confidential?

HR A: Appeals can only be seen by the HR manager, so it's fine.

SD A: This is capability; this can be seen by the line manager.

HR A: Oh really! So HR B you're right we do need a confidential tab.

SD A: Let's change it then!

Grievance comes in and doesn't go out, I wonder if it is the right term?! Grievance Response Sent Date.

HR A: Yea, it would be grievance received, and then response sent and then the date. An appeal would have the same process, just that there is an appeal date.

SD A: [Explains forms and things that need to be added].

Is there a penalty to a grievance?

HR A: No, not really. Unless it was a disciplinary case in account to grievance, then I suppose it would all be wrapped together, but not really no. I would keep that in the disciplinary section.

SD A: Would you have the penalty?

HR A: No, the penalty would start the disciplinary process. Wait no! There might be a penalty later during the process, they might merge within.

SD A: [Explains disciplinary action stages].

HR A: Can we change the topics within?

SD A: Yes, you can

[Explains disciplinary action].

How many stages do you have?

HR A: 3.

SD A: I think that's standard. The key thing we want there is on informal you lose termination date and suspension date.

HR A: Well really, informal....

SD A: Verbal warning...

HR A: The stupid thing about a verbal warning, is it can't be written down anywhere, because then it is no longer a verbal warning. So if we wrote it down in here, it would no longer be a verbal warning. It's ridiculous.

SD A: So we had an informal conversation?

HR A: Yea.

SD A: [Explains disciplinary].

HR A: On the other screen, investigation, allegation should be the other way around

SD A: Well spotted. [Explains disciplinary].

HR A: Why does it say companion twice?

SD A: Can you have 2 companions?

HR B: You can have one internal and one external.

HR A: It should say that then.

SD A: [Explains disciplinary]. Do you want a dates notes tab?

HR A: I am pondering on what is the difference between this screen and the disciplinary / grievance screen. On the appeal page did it say what the appeal was. We need a tab that shows what is appealed.

SD A: Are you going to have a chair of appeal investigation?

HR A: I don't think you would have a new allegation, but you would have a summary of the appeal.

SD A: [Discussion what a penalty is].

HR B: So when you have a first outcome, when new evidence comes to light, what is the new outcome?

SD A: Yea. So that's D&G and the appeals that sit within.

(Date 4.3.13)

During this session disciplinary and grievances pages are discussed. The pages are examined and each of the categories presented discussed whether they are needed or have appropriate language. Confidentiality is central in disciplinary and grievances since only appointed people within the human resource team should be allowed to see the information. Verbal as well as documented warnings are discussed and their legal impact. Extracting knowledge through socialization enables internalization to take place for each group member. This then allows a new group tacit knowledge to surface. The topics go into more detail as the conversation goes on and allows the spiral of knowledge to prosper. Constructive learning is one of the main form of exchange in this discussion. SD A explains the pages, and HR A and HR B are able to react to the information provided. This triggers conversations within the group, which allows group tacit knowledge to prosper and evolve. SD A is also able to use tacit knowledge transferred from HR A and HR B to complete the pages and understand the procedure of a disciplinary committee. The visual triggers from the page spark socialization. Seen in Tables 89 and 90.

Table 89 - Extract 42 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	x

Table 90 - Extract 42 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	x	x		

Extract 43:

SD B: Things that are a nonstandard you can set up as a process. So you can add in a 2. probation. So, you can't have it as a field on the employee record. You cannot see an employee list with ticked probation. So the question is whether everyone can see the second probation or you set up a process, where you can see that everyone needs to do a 2. probation step, by a certain date.

HR A: Do you go through the employee to get to the process?

SD B: There are various ways, you can through the dissimilarities etc.

HR A: Since they employ 30 people at a time, and they all need to get their review taken after 3 months. We do need to pull up searches and lists and see whether that's been done or not. So we need a field.

SD A: I will get you both a login, so if you want to go and play around with the system, I would advise you to go into the Demo system first. Once you are more confident you can go into the customer's system.

(Date 5.3.13)

SD B shows the process of reviews and probation, which has several steps. HR A is trying to understand the procedure how to get to the probation steps as well as help SD B understand the needs from the HR side to the software. In order to understand and find out the needs to of the HR team, logins are then provided in order from then to use the software and see what they require. Constructive learning, externalization, with visual triggers leads to socialization seen in Tables 91 and 92. This allows a process of internalization to take place for the group and enables a new group tacit knowledge. Finally, the tacit act of allowing the human recourse consultants

to use the system enables a spiral of internalization through visual tacit triggers.

(Date 4.3.13)

Table 91 - Extract 43 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	x

Table 92 - Extract 43 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	x	x		

5.2.7 Project Plan

This section focuses on the delay of the project due to several issues with the project. These mainly include the transfer of data from the old to the new system. The executive decisions made are crucial for the project and its success.

Extract 44:

HR C: I am quite happy from where we started and where we are now and what HR A has achieved. I just want to make sure we meet those deadlines. I prefer to do it properly, rather than everything has to be ready for the launch.

CL A: We already agreed, that we would not do everyone in one big bang implementation. The idea was to roll it out at the top first, so the SMG

has experience with it before we rolled it out to the managers and line managers.

HR C: CL A can I ask you, if there is any reason why we are not looking at training within the next 2 weeks. And a roll out at the 1st of April, do you have any.... are you happy for it to roll on a little bit further?! Is that what you are saying or would you rather be trying to work with those deadlines?

CL A: I'd rather it was right, what I want to make sure that doesn't happen is that we stick with those dates if we can't confidence the materials that the materials in place are understandable and thorough between us, we haven't come up with a plan for the training. So I think that a really useful step is to move away from the not very good system that the museum has and do it properly. So I can afford to push those dates a little bit.

HR C: We will come back to you tomorrow and confirm a date.

CL A: So if we went live on the 1st of Mai, we could have the training in the first week, just before you return (HR A on vacation till end of April).

HR A: I think that would be better. Like that I can pick up the training at the end, after you start, so I can learn how to train.

SD F: So we can do the 1. and 2. of Mai.

SD A: So we can go live on the 1. of Mai, train the 1 and 2 as well as the 7. We can also launch the Webcasts.

HR A: So we can delay the payroll by a month as well and give us the 2 months we needed to get that ready.

SD A: This would also give us the time to see CL A and give her an advance notice of what is there. So that if she is raised any questions, she has an inside view of the system.

HR A: So maybe mid-April.

SD A: Until we fixed the functionalities, the screens may change. Until the screens are right we can't do the training videos.

[Explains project plan].

(Date 11.3.13)

This conference call mainly revolves around when the system will be implemented in the organization. Training is needed for the employees, so they are used to using the system. Socialization allows SD A to explain where they are in the project and the time that is still needed in order to finish the work. This then results into a discussion between the executives where the date of launch is pushed back to allow a more feasible system to be delivered to customer. At the end of the discussion a new group tacit knowledge is created by setting new dates for the project, in summary Table 93 and 94 show the tacit knowledge.

Table 93 - Extract 44 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x		x		x	

Table 94 - Extract 44 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
	x			

Extract 45:

HR C: Fair enough if there are delays. I tried pushing CL A into that direction yesterday, but with the delay we have, I want to make sure it is the only delay we have and then thereafter it will be fine. I wanted an Overview on what we have done, what we still need to do and how long it will take with them.

SD A: The issue is for us is that every amendment that we have to do has the same priority order, so there are things which are still being delivered and tested here. There is also a brilliantly helpful stuff from HR A and HR B saying that doesn't make sense, can we move that here, which isn't big stuff but we need to deliver it to the customer without the screens changing. The other thing is that our payroll system is very different to the old system.

HR C: How different is it? The old system generally pulls everything vital in for us except for sick pay and absence.

HR A: The way it pulls in the data. We thought we could just pull the information off the old software and put it into NetSuite but we can't, we need to jiggle it around and reformat it. Find start and end dates that we didn't already have. There have been quite a few glitches with the paper one. Even the way it pulls it out is quite different.

SD A: So now what paper does, just throw out CSV files out of paper, NetSuite doesn't do that?

HR A: It does, it's just the format.

SD A: The two things that need to be spot on at roll out are holidays, because that's the first people look at and pay. We assumed that pay won't happen till the end April anyway, but then we found out that they have weekly payrolls. So the roll out is in April May, it's important that the things are spot on for senior managers. I think we can't do it in a fortnight since we still got all the data. Before we don't have the data and the screens right, we can't do the training materials, since they will change.

HR C: What are we looking at in terms of weeks?

SD A: Well, I think if we take another month, so we go live beginning of May instead of April we should be there.

HR C: I think the delay is quite good, because you are away (HR A). We are replacing what they haven't been really happy with, so consequently it

is a bit of a sales job as well and I am easing it into them, making sure that they are really happy with it.

HR C: And CL A still hasn't signed off on contract yet either.

SD A: Exactly. So if I go back to CL A today, I will tell her we had a brief meeting, it's pretty much on target except for the payroll and the holiday, i.e. calendar, is that right?!

SD A: Yea.

HR C: And that we are now pushing the start date to the first of May.

HR A: And we will revise the training dates.

SD A: [Explains NetSuite Database].

(Date 12.3.13)

The extract above demonstrates a discussion between the HR consultants and the software developers, regarding the discussed time change of the project. The new delay by 1 month, as well as the training allows the amendment team of the software to finish the current problems. Most of the problems which need to be sorted out are tacit tasks, such as restructuring the format for the payroll, in order to feed the new system. Socialization between the HR executive and the software development executive allows internalization to take place and new group tacit knowledge to build. The discussion results in constructive learning, externalization, by SD A whom explains the system to HR C enabling the system knowledge and issues with the transfer to internalize. Tables 95 and 96 show the tacit knowledge categories.

Table 95 - Extract 45 Results Tacit Knowledge

Socialization	Externalization	Internalization	Combination	Group TK	Constructive Learning
x	x	x		x	

Table 96 - Extract 45 Results Tacit Knowledge Triggers

Visual Trigger	Conversational Trigger	Constructive Learning Trigger	Anticipation Trigger	Recall Trigger
x	x			

5.2.8 Summary

The extracts above showed the occurrence of tacit knowledge during the meetings. It was demonstrated that socialization, internalization and group tacit knowledge was found in all 45 extracts. Externalization on 28 accounts, combination in 9 and constructive learning 18 times. This established tacit knowledge exchange in the project.

Following from this, tacit knowledge triggers were found throughout the extracts as well, which established what can make tacit knowledge surface. Visual Triggers were found 18 times, conversational 39, constructive learning 19, anticipation 2 and recall triggers 7 times. These triggers will help establish the model in the following chapter.

The previous examples aimed to demonstrate the exchange of tacit knowledge in the software development project. These were analysed through the theories of Nonaka, Ryan and Clarke. The use of constructive learning, socialization, externalization, combination, internalization and as a result obtaining group tacit knowledge were highlighted. Clarke's triggers gave way to further analysing when tacit knowledge surfaces, these will be discussed in further detail in the next chapter. The meeting extracts and its analysis allowed a further understanding of tacit knowledge transfer. This will aid in the investigation of decision making and its relationship to tacit knowledge.

5.3 Decision making and tacit knowledge

In the previous section extracts of the meetings demonstrated the existence of the elements found in several theories such as Nonaka and Teece or Ryan. After having highlighted their existence and the interplay between the different types of tacit knowledge found in during the meetings, the focus of the next chapter will be the impact of tacit knowledge on decision making.

Using the NDM model each extract is evaluated accordingly and demonstrated through the model at the end as well. The key decision maker and which project member influences the decision supports the tacit knowledge usage of project members to solve problems. This will aid in building a basis of data which will later be evaluated in the next chapter.

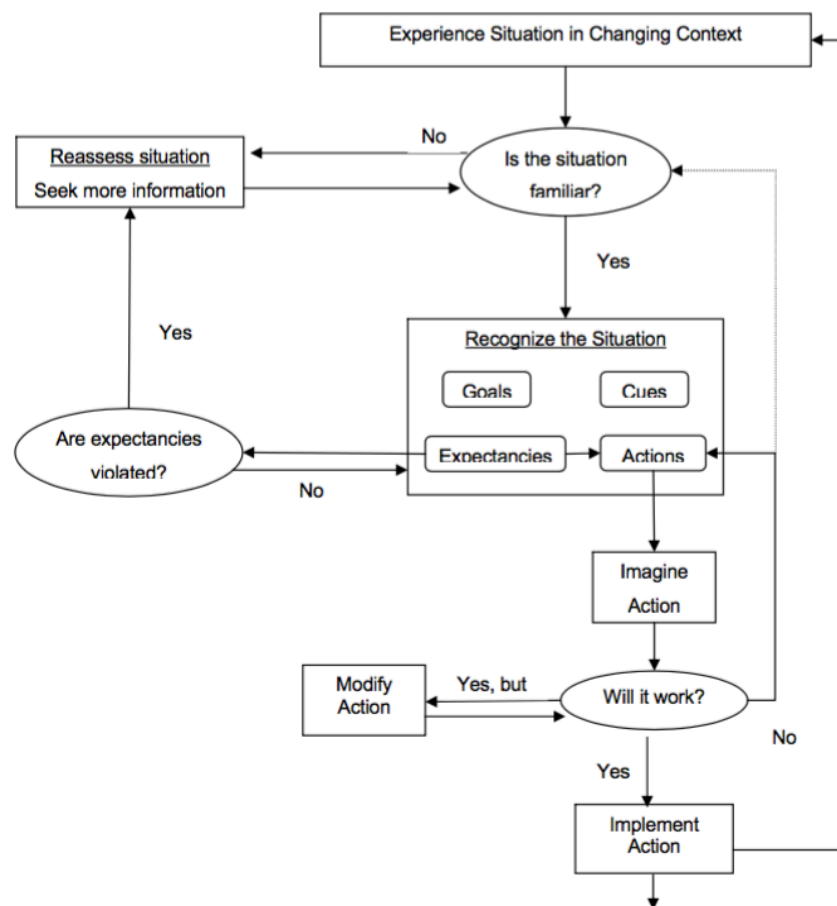


Figure 15 - Recognition-primed decision model (after Klein et al., 1989)

5.3.1 Decisions

In the following section, the decisions made during the meetings are demonstrated and analysed. They demonstrate how project members use their tacit knowledge to contribute to the team and make decisions. The member with the most expertise or seniority is at the centre of the decision making and hinges on the knowledge of others to achieve results. Each extract will be shown through the recognition-primed decision model.

Extract 46:

SD A: So Payroll, while you mention that, we would really like to not switch off the old software and go live with the new payroll programming one hit, because from the HR side that's not an issue at all. But payroll is so much more important and our payroll is a bit different from the old software thing. So we would like to put that off for a few weeks, just to make sure that the data matches.

CL A: That's ok.

HR A: SD A, can I ask a question about payroll. I completely understand what you are saying about the old software. Would it be best, bearing in mind that we are coming up to the end of the tax year to leave the old software running before we switch, or is that not going to make a difference?

SD A: I think it would be better to leave the old software running on the payroll side of things.

HR D: Correct, so we shut off at the end of march for that year end and then start the new payroll software beginning of April.

SD F: No, that's too soon!

SD A: No, because the go live isn't till the beginning of April anyway, so we won't have anything to match.

HR D: Right.

SD A: What we are dealing with live, is that we can generate the payroll report out of the old payroll program and generate the payroll report out of the new software's payroll and confirm that everything is accurate, before we just switch off the old one.

HR D: So what date were you thinking to make the final switch?

SD A: Well that's something to talk through with HR A and HR B, which we will hopefully get to this afternoon. Possibly it will be after the first monthly one, because we won't check the monthly one till that's done. We will go through this, this afternoon, whether after checking the weekly for a couple of weeks, will give us sufficient confidence.

The payroll is a big thing; I didn't realise it is a weekly payroll.

HR A: Even if it was a monthly, which I think goes out on the 25., they prepare it on like the 11.

SD A: If the old software would come out earlier, we could match it.

HR A: We have to keep in mind as long as they are running on the old software, they are paying double. I was surprised that CL A was a calm about it as he / she was. I thought he / she would ask about cost.

(Date 4.3.13)

This extract portrays the project's evaluation as whole, where the change from the new to the old software is discussed. SD A who is the most familiar with the problems of switching from one software to the other is asked whether it is feasible to switch off the old software and go straight to the new one or keep both running and compare the outcome to make sure they match before switching. The need to keep both programs running is a decision made or suggested by SD A due to her familiarity with the subject. SD A recognizes the goals needed for the success of the switch, comparing the new and old software payroll sheets and see if they are accurate. HR D's concerns with the costs of running both software also comes into play as well as extra time needed until the switch. Due to a previous conference call, the expectations are not rejected since the customer did not oppose the

decision. The action is then imagined, or planned, by SD A, HR A and HR B in more detail, looking at the form in which payroll presents itself and when they will implement the action in order to achieve the goal, seen in Figure 15.

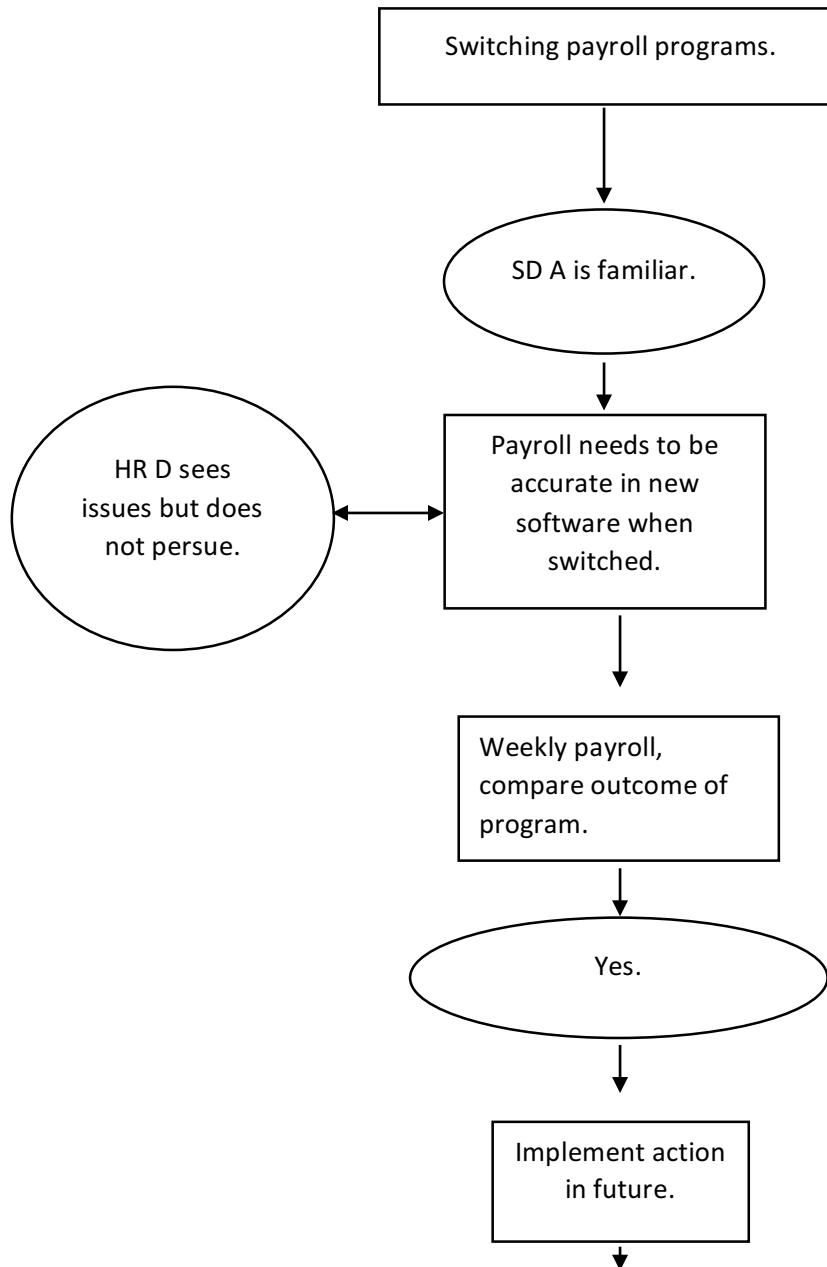


Figure 16 - Extract 46 Decision Making

Extract 47:

AC A: The pensions are the problem because obviously we need to work out the formulas with the customer. It depends on the percentage; do they get 3%? There are other percentages depending on the salary, we take a lot of time working out what the pensions are due to the formulas. We need a specific Figure, so we know what to put in, so we don't have to calculate the formula. It is not a normal Pension, they calculate on salary sacrifice and all the employees which are not on standard salary rates.

SD B: So you need the percentage contributing from the employee and the percentage of salary sacrifice and from those you can calculate the actual percentage which they are contributing and the employer is contributing.

HR A: What I can do is show a spreadsheet which CL A sent to me a while ago to SD B so he/she can see what you are trying to describe because it is hideous. It is easier to understand when you see it. It's all to do that their salary sacrifice comes of their salary tax and pre lots of allowances. It is quite complex.

SD B: So if we come up with a sample spreadsheet to send you, with all the Figures in, which are relevant. You can approve it.

AC A: Yes, that would be best. This way we can say what else we need.

SD B: So to summarize, we have the employee spread sheet, with generic information and addition and deduction types. The employee addition and deduction sheet. Employee pension spread sheet and attachment of earnings.

AC A: AC B has asked CL A for a pension spreadsheet, since we always hit a wall when calculating the percentages of pensions. They want more information on their payslips for mortgages etc. So we are working on a new payslip with CL B.

(Date 5.3.13)

Pensions were one of the main issues during the project due to its complexity. Explaining pensions was confusing to the team even after several attempts. In addition, certain information has been missing on the payslips of the employees which should be part of the future payroll payslips. A new, accurate, spreadsheets needs to be developed to feed the new software with the right data from the accountants. At the same time a sample spreadsheet needs to be developed by the software engineer to give a form in which the data needs to be fed into the system. The decision to create a sample spreadsheet for the accountant is given by the software engineer, but additional information needs to be added by the accountant. The customer needs to sign off on the sample spreadsheet and add information if needed.

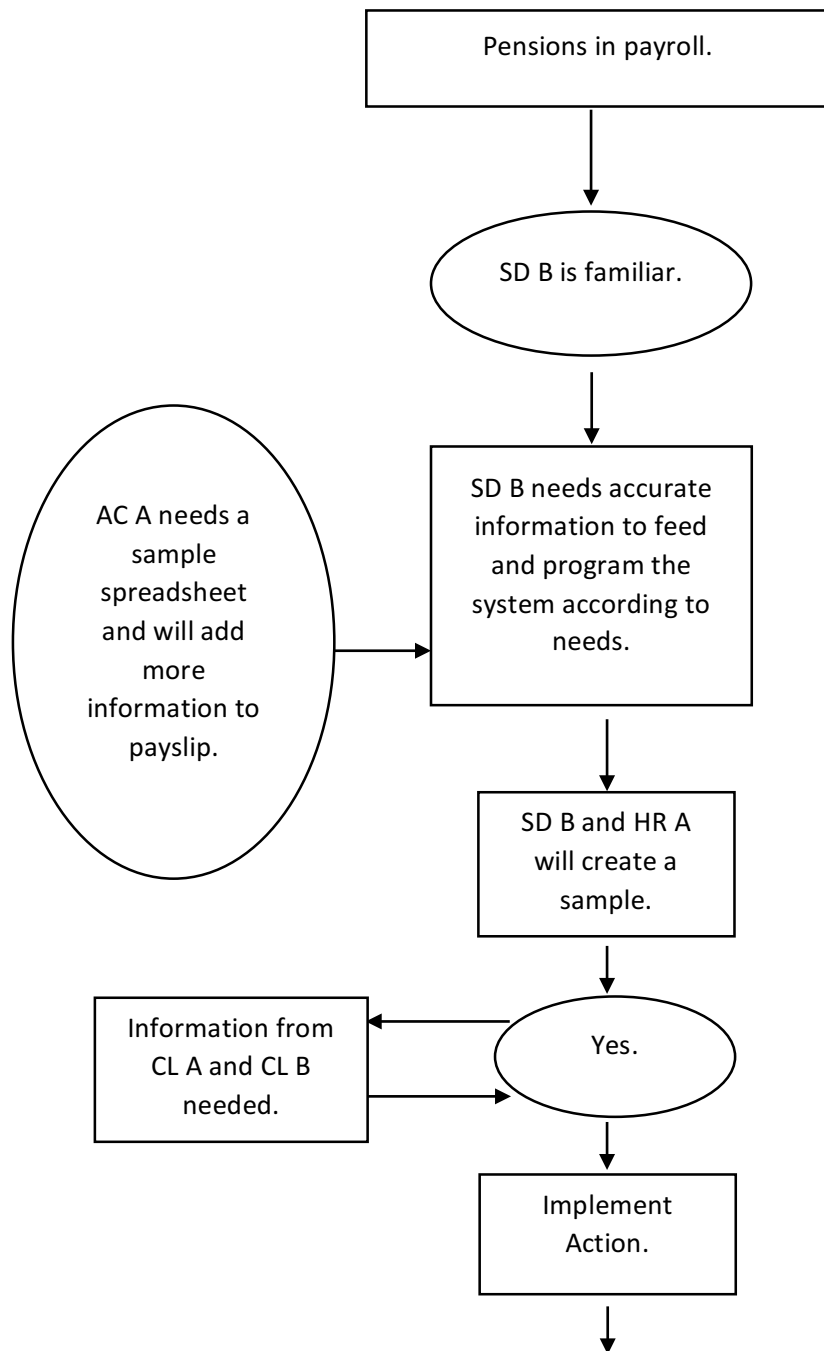


Figure 17 - Extract 47 Decision Making

Extract 48:

CL A: I have a couple of questions regarding first the historical absence data. Can we put those in?

SD A: Yes, you can put in a historical, current or future absence.

CL A: That's really helpful. There is another thing, concerning what you just said SD A, we want to see an annual payroll review, probably through paper which means that effectively will be changing the pay during April. All this means really is that when we take the final data, we don't take it at a point where half the changes are done and the other half isn't. That would be a nightmare.

SD A: On the payroll, I assume you want the current numbers and the old ones. So effectively you would want last years and this year's payroll numbers?!

CL A: I think we would need both from an audit point of view.

HR B: What bothers me is that there is so much room for error. I feel like I should just be putting it straight into the data base, rather than trying to put it into a spreadsheet first, so I can look at it.

SD A: Why don't you do that?

HR B: At the moment, I am doing additions and deductions and pensions, like I said before I'll give it to you but I need to go through CL C, I was basing it on the reports, but it doesn't match. AC B has some stuff that we don't have, we have some stuff on paper that he is not paying so is that because there is an end date that should have been put in or is it not relevant anymore?! Is it just a user error for not putting in an end date or is there stuff that should be paid but isn't being paid?! Do we need that to be resolved before the data goes in? I think it does, but how long is that going to take?!

SD A: I think we need to get data in there, it's easier to go through and say these 70 people don't have this or that, than to be waiting for perfection, when we have 200 people to check.

HR B: We have everything that is in paper but someone needs to go through and check what is right.

SD A: CL C?

HR B: He / She is the only one that can do it, but if we wait for him/her to do that, we will never get in, but if we load all of this in as being current, people will be getting things, that shouldn't get anything at all.

SD A: How about scheduling a call Thursday morning with CL C and go through it with her, so it doesn't go to the end of her pile.

HR B: You think she can answer that like that, or does she need to look at every PM?

HR A: I think she might have to look into them.

SD A: If we email it today, and schedule it on Thursday, she has some time to look over it.

HR B: What we are dealing with are records not being closed.

(Date 25.3.13)

Payroll is once again the centre of discussion in this example. Feeding the software with historical payroll data is crucial from an audit point of view. Using the fed data to analyse the different aspects of payroll gives a more detailed view for the organization where the main expenses lie over time, as well as giving the employees the opportunity to see a detailed view of their income. The expert knowledge from each participant aids in the decision-making process. CL A shares the needs of the enterprise, where the auditing of the payroll is important, HR B combines it with the knowledge of the payroll day-to-day job as well as the software and how the data will be fed into the system. The decision lies within HR B in how the historical payroll data will be handled with respect to the clients wishes. Combining the gained knowledge throughout the project HR B is familiar with the situation and can find a solution to progress the historical data payroll input. SD A's initial proposal to put the historical data into a spreadsheet and feed it into the system is opposed by HR B due to the margin of error which can occur

during this process. HR B decides to put the data in directly which is agreed by the customer as well as the software developers. The knowledge HR B uses to solve the historical data issue is a combination of new software knowledge, old software knowledge as well as the payroll data of the customer. This process is shown in Figure 18.

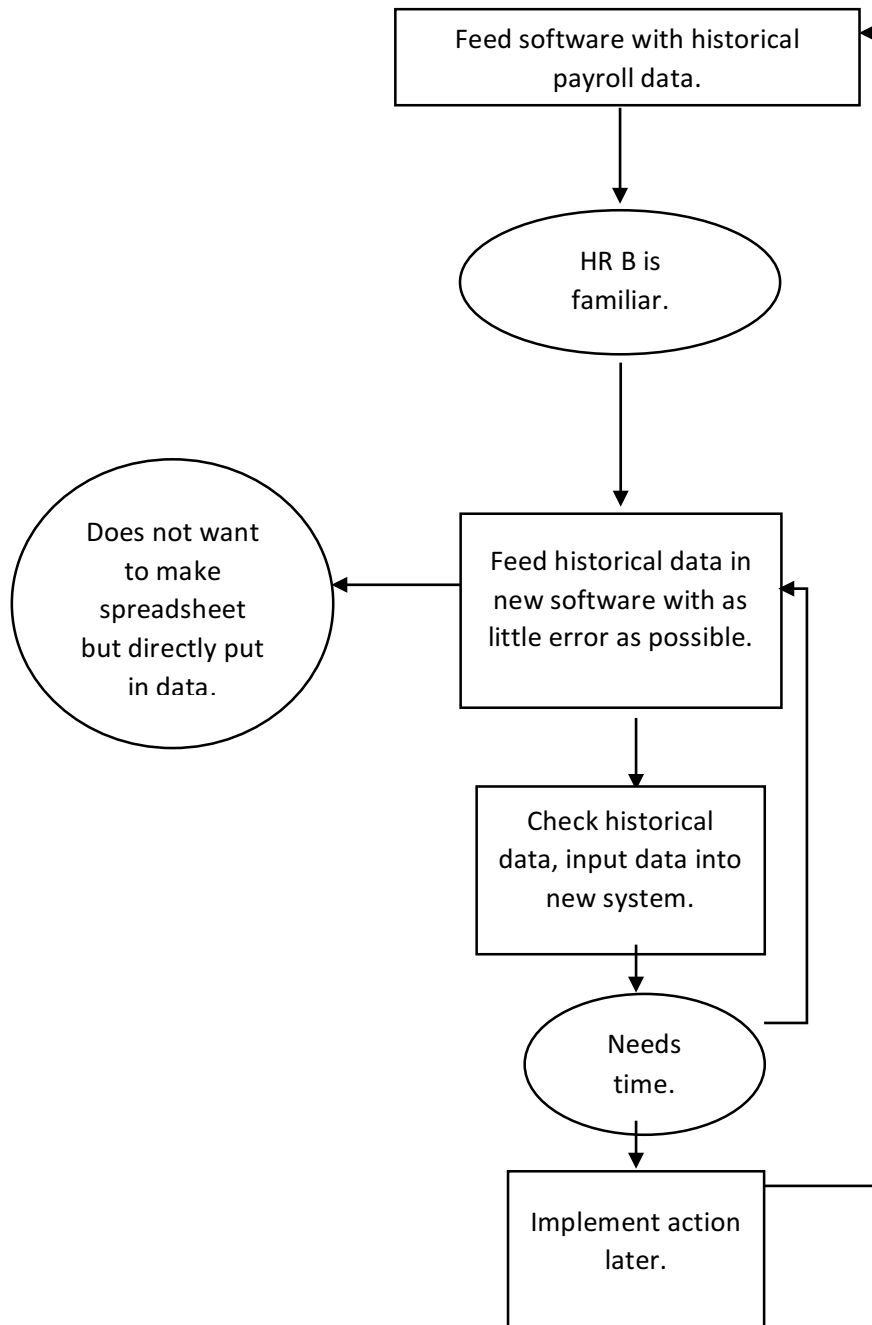


Figure 18 - Extract 48 Decision Making

Extract 49:

SD A: Now we are getting into linked records, we have done the core records. We talked about name changing, to be the item type. Appraisal type. Standard appraisal, 360 appraisals and a scoring appraisal. So this is something to look at with SD B tomorrow.

HR A: My thoughts on the whole is that we will probably have to change some of that, but I am not quite sure to what yet, until we start building the form, and then work through every stage of the process. I think it will become clearer.

SD A: Is there something from the old software that could make it clearer?

HR A: No, because they currently don't use it. I've got draft one of the questionnaire done now, which I would be happy to send to you but it hasn't even been checked by CL A yet. While we're at it, you know we talked about the summary of the feedback and SD B asked what kind of format do you want it in? We just got some of the internet that CL A quite likes, do you want them now or should I give them to SD B?

SD A: SD B. The feedback is in the process engine, so that's his / her part.

(Date 3.4.13)

The customer wants a 360-feedback system for the organization. Using a matrix system for each employee, questions are generated to see whether an employee is happy with their manager and / or the people they manage. The software organization generated a system where each employee can go through the questions and the results are sent to the HR team. HR A sees the system created by the software developers and decides that most of it needs to be changed. Utilizing tacit knowledge gained through experience in the job as well as the needs of the customer, the decision to further the conversation with the customer present is made. In addition, the formatting of data which needs to be fed to the system is still unclear at this point. Due to expectations being violated, the 360 feedback is put on hold till further knowledge is gathered. In the following section, recall

decisions, the 360 feedback topic will be picked up again. Within the NDM model the process is shown in Figure 19.

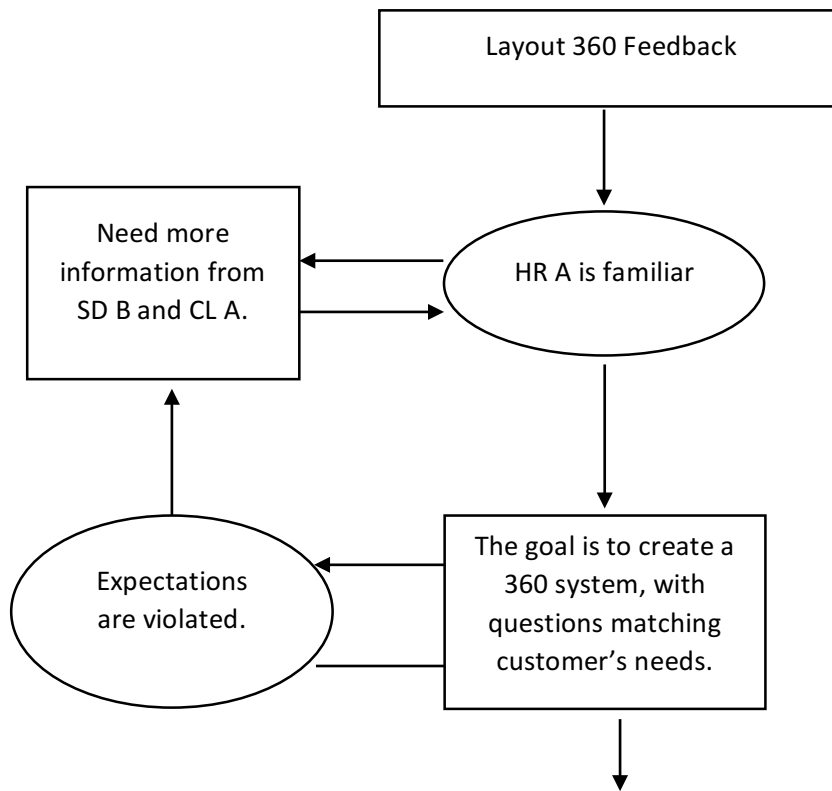


Figure 19 - Extract 49 Decision Making

5.3.2 Recall Decisions

The focus of this section are decisions made at a previous time during the project but were forgotten. Through discussion they surface again and are used to advance in the project. Unlike the previous section where decisions are made during the meeting, here previously made decisions surface and are discussed.

Extract 50:

HR A: The 360 stuff that you sent to me earlier, can we run through it now?

CL A: Sure. [Explains 360 questions.]

HR A: [Explains 360 questions, how they were made, categories and system.]

SD A: If you could give me those questions in a spreadsheet that is cluster – skill – question and a line, then I can pull them in. You can then pull them into question sets and set up some feedbacks. And then we can write responses, and give an example to CL A.

HR A: yes.

SD A: [Shows software – 360 pages.]

Will you use the same question set across multiple categories?

HR A: [Explains matrix of 360 questions.]

SD A: How many different question sets do you have in your excel spreadsheet?

HR A: There are four, what CL A calls categories of staff.

SD A: How many of those are going to get the same question set?

HR A: None...well there will be some duplicates cause for example the skills questions, are going to everyone, whereas the business leadership sets of questions are only going to 2, the senior managers. How does the system know what they are? (Employee, senior, line manager)

SD A: Cause you would do a search to find all the senior managers?!

HR A: It's not as easily categorized as that sadly. We know, because CL A gave us a list of each but...

SD A: So if we go back into the system, in employee record.

HR B: Is there an indicator that we could use on employee record? To categorize them, give them a reference number. Such as senior manager is category 1, line manager 2 etc.

SD A: It could be grade, I was thinking about taking it off that excel spreadsheet, and that putting each column as a grade 1,2,3 unto 8. So 1 was the top. Would grade do it?!

HR A: If we called it.

HR B: Is the feedback anonymous?

SD A: I thought it wasn't, but we know who we are sending it to anyways.

HR B: If it was, we could link a number to each employee and then reference their position with it.

SD A: I think if it is anonymous it would go a bit under the fence.

HR A: That's what I think, but CL A isn't sure about that at the moment. I think she would like to have the option.

SD A: I mean you could take the name off; the thing doesn't go out anonymously because we know who we are sending it out to. So it's just a matter of how you present the results, by taking the names out.

HR A: I think if we could draft that both ways. So if we stagger over the year, I think we are going to start with SMG first. As a test group, it would be helpful to have a list, who are the reviewees for May or whatever it is.

SD A: [Shows how to send a feedback in software.]

(Date 11.3.13)

In the previous section, decisions, the 360 feedback was discussed. However, due to lack of information from SD B as well as CL A the discussion needed to be put on hold. HR A has gathered knowledge from CL A to further the 360 feedback pages. Recalling decisions made by CL A on how to present the feedback allows tacit knowledge to surface. HR A asks SD A to present the feedback anonymously as well as with names is due to the uncertainty CL A has with the subject. Some decisions have yet to be made in accordance with the needs of the organization. HR A is the most familiar with the situation, having gathered knowledge from CL A as well as knowing the topic from previously gained experience. Tacit knowledge can surface, as well as the decisions made by the customer and the HR consultancy to further the 360 pages. The goal is to have the questions in clusters and presenting the feedback in an appropriate way. The software development team has offered a first draft to present the feedback which is

then altered in accordance with the needs. Seen in the process below, Figure 20.

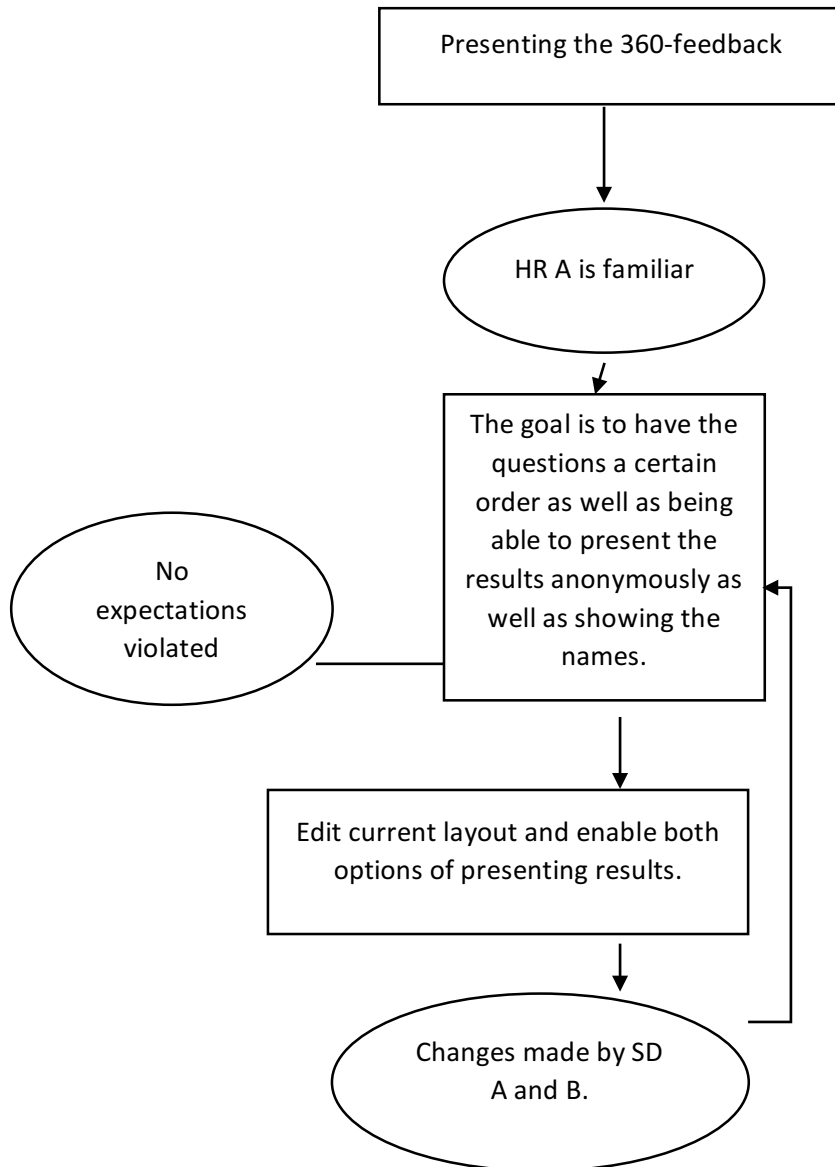


Figure 20 - Extract 50 Recall Decisions

Extract 51:

SD B: [Explains categories.]

SD A: Just something that CL A raised yesterday, is that they have a very rare occurrence of agency workers. So, I just wanted to show you the list, where you see the range of options you've got. There are three sorts, employees, apprentices and shareholding directors. So, I wondered if it is at the right place. Is it used as an employee search at all?

SD B: I think it is in one search but not generally. It can vary, they can start as an apprentice, become an employee and then a share holding director.

SD A: I think it was the Employee Record not the Employee Centre, that we will put in the last 3 tabs.

HR A: Yea, that's what I've said.

SD A: Sorry, I just went back.

HR A: Oh no no, we started at the one end and then we went to the other end. It was the 3 in the middle.

SD B: Should they see their tax code?

HR A: I think they should see their tax code and national insurance. When you get a new tax code notice, you should give it to your employer.

SD B: And you NI category?

HR A: I don't think people know what that means, so no.

SD B: What about checks, verifications? We could put in check the visa but we can also put that elsewhere. I hope it is not confusing that it covers all verifications.

HR A: I think verifications is the best option.

SD B: The question of changing position to post. If we've called position, it will link to the organizational post and then we've got post start and end date. So, I am wondering what is the difference between a position and a post?

HR A: We discussed the topic with SD A, and she explained the process, and the difference between the two and I said we need to name them different.

SD B: The problem is that here you link them together where they are in fact the same.

HR B: I thought that it wouldn't be the post that ended but it would be the position for that PM that ended but the post is still there. So for the organization the post is still there.

SD B: So we need to change that to position start and end date.

HR B: The employee doesn't need to see the post information.

SD B: Let's just hide that.

HR B: Where is the post information held?

SD B: In the post records, under organizational cases.

You can create process for customers, sales leads, employees.

Is there anything you have to do within a week or month of them (employees) arriving?

HR A: Contract has to be issued within 8 weeks.

SD B: I presume there is stuff you have to do before they arrive?!

HR A: Yea.

SD B: So, the first thing you have to do is process 1., then the key date name/start date. So, then you set up stages.

HR A: Send a letter to the employee. Send acceptance

SD B: Should we say that these people already agreed?

HR A: Send CRB form

SD B: prepare desk.

HR A: notify IT

SD B: Let put in mid-way review, which is the next one in the sequence.

HR A: What would be great is when you set up the real customer's engine, you could give us access so we can start building the processes.

SD B: Once the steps are set up, we go to an employee.

HR B: Can you start up a process automatically? Like when I put in a start date, will it automatically launch the process?

SD B: You'll see the list of things you have to do. Suppose we had create a disciplinary template, we clicked on create process steps. Then it inserts all of these fields for you.

HR B: Oh no, I mean when I start a process will the process template be attached automatically?

SD B: We can't link it. [Shows process.]

It's probably best that you try it out yourself.

HR A: Yea, it's one of those things where I feel like I want to try it out.

I have a question; it has just fallen of my head. SD A told us yesterday that NetSuite have provisionally agreed on the Buro setup. That they still have to thrash out some details but it looks promising. So say we had that setup for us, the HR consultancy, and for each of our clients. How would the process steps work then? Could we set them up at both levels? Like generic ones for each of our clients and general ones as well?

SD B: What we are building is an import/export process so you can transfer the exported data from one account to the next.

HR A: So, if we do best practice templates we could export or import them to each client.

SD B: Yes.

(Date 5.3.13)

The extract begins with the explanation of categories of employees within the organization. Throughout the conversation several decisions are made in relation to the employee centre, which holds all the information concerning an employee and their status within the organization. Knowledge is shared from HR A, HR B, SD A as well as SD B which allows changes within the software to be made swiftly. The range of knowledge during this meeting is quite high due to the software developer being present and changing the software in accordance to the needs and knowledge of HR A

and B. During this process HR A and HR B recall decision made with CL A to what specific needs the client has with the software. The process of looking at the software and asking for alternations can be reflected in the NDM. The situations are familiar for the project members, however expectations from HR A and HR B are at times violated and alterations need to be made within the software. SD A and SD B are capable to make these alterations, at times instantly or in the near future. The knowledge surrounding the discussion is exchanged, recalled and passed on from one project member to the next, to create an employee page in accordance to the wishes of the customer. Figure 21 below shows extract 51 in visual form.

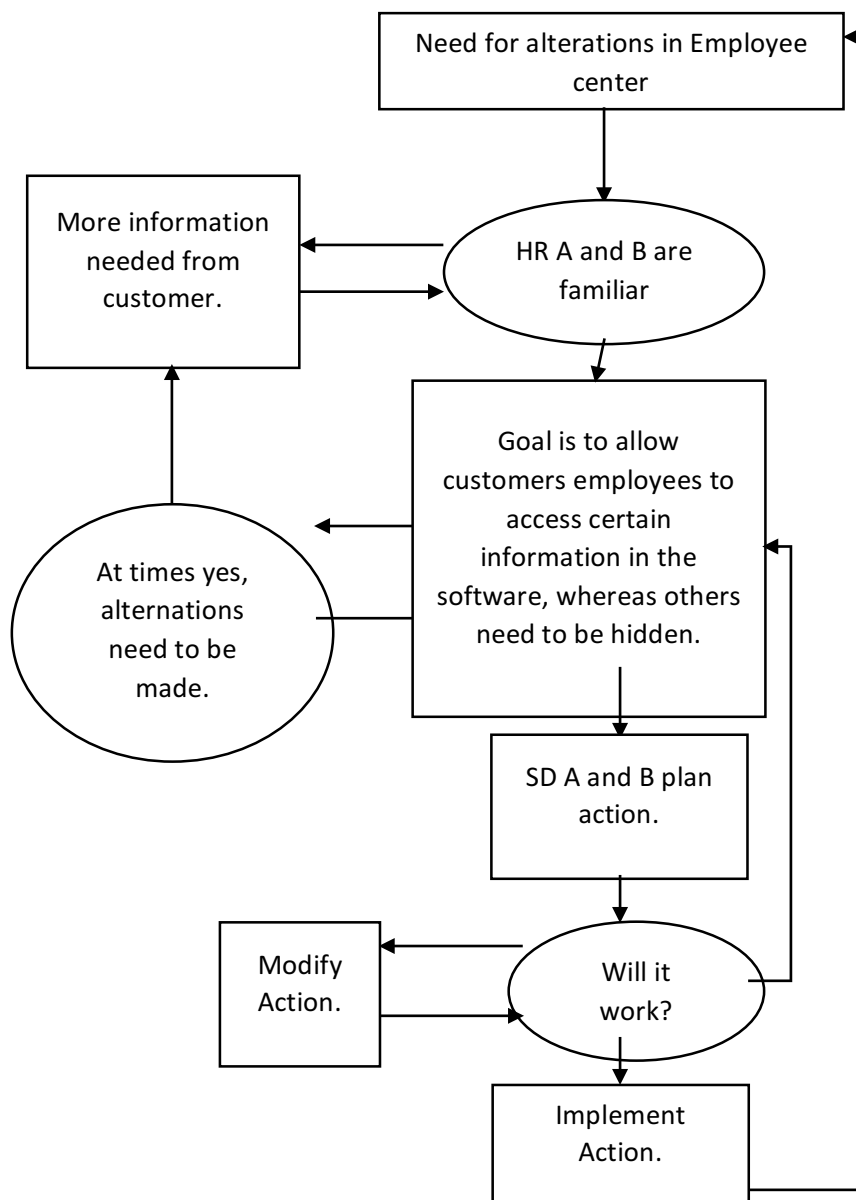


Figure 21 - Extract 51 Recall Decisions

Extract 52:

SD A: [Shows Employee Centre.]

HR B: That's weird, I was doing this post. I was in the middle of the record and I put in the information and then I pressed save and then it said I need a salary. Then a pop up come and asked if this is his substantive post and then it took me back to the employee screen, it didn't let me put a salary in. It took me away from where I was.

SD A: Either, we haven't got a working pattern, and that is tripping us up, or we have 2 things going on and they are tripping us up.

I just went back and there are 2 things that SD D is working on related to posts.

HR A: We need to talk about the wording.

SD A: What would you like?

HR A: The data we have given to SD B, is some is on a need to know basis, so just the HR team and CL A and then there will be some which the line manager needs to know.

SD A: Equality and Diversity is going to be confidential information and the other is going to go into the verifications.

HR A: Is a chronic asthmatic, for disability and discrimination purposes that would be considered a disability because it is a lifelong problem. So, it's the kind of thing I would tick that new disabled box for and it would also fall into medical alerts because the line manager would need to know. Or, could we add it to medical alerts? Does this medical alert have to do with disability, or something like that?

SD A: And then have a housekeeping check. But if they are not registered in E and D, does it count in E and D?

HR A: If it is something that affects you doing day to day activities it is a basic disability. For example, if you have cancer are you registered disabled?!

SD A: No.

HR A: Cancer and HIV are the 2 named exceptions under DDA, as soon as you are diagnosed with either of them, you are automatically covered by E and D.

(Date 11.3.13)

This extract also focuses on the employee centre and the accessibility of information within the software. In focus are the medical alerts, where at times line manager need to be aware of the condition as well as at times it needs to be confidential. A medical condition can also fall into equality and diversity, which is the case of cancer or HIV. Due to the customer not having a software for medical alerts, the old data needs to be checked by the team to verify the information provided by employees. In addition, the wording needs to be changed in accordance with the HR consultant's vocabulary within their organization. HR A asks for clarification from SD A as well as changes within the medical alerts in accordance to their internal vocabulary and standards. This can be seen in Figure 22.

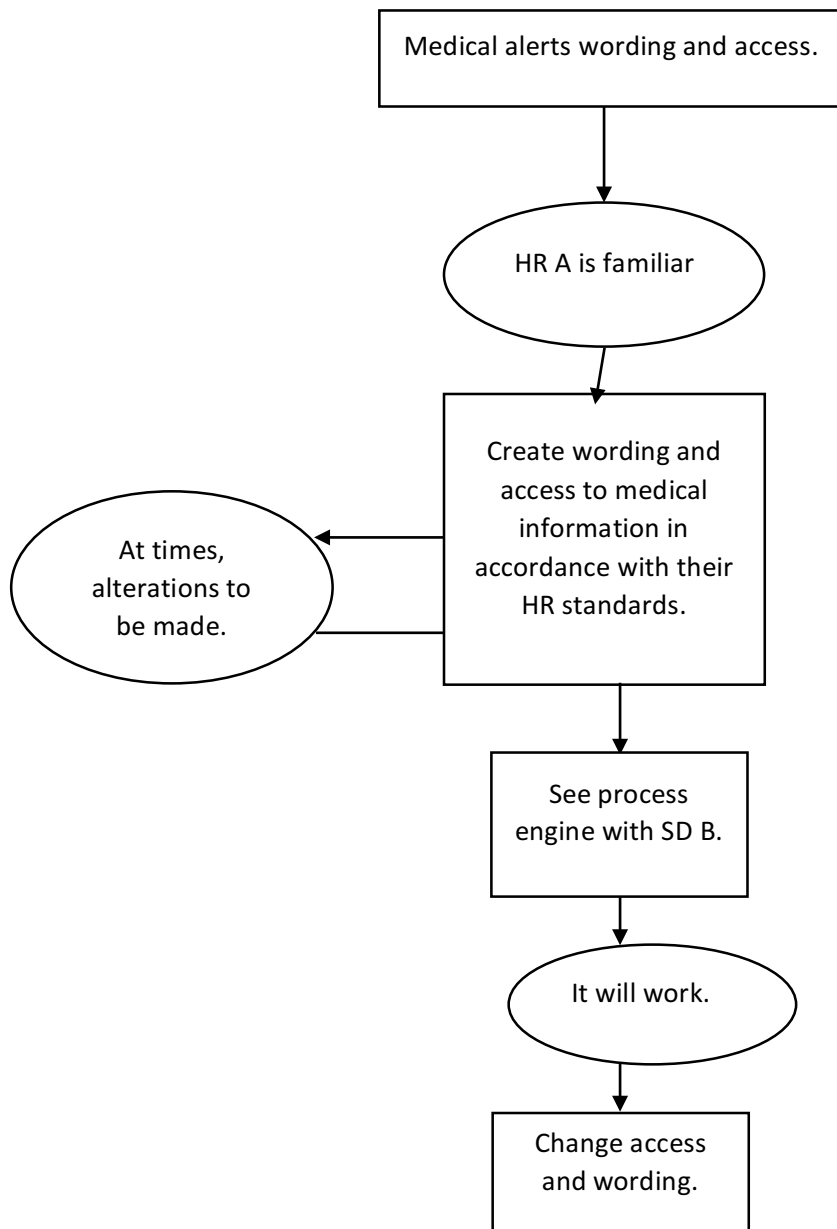


Figure 22 - Extract 52 Recall Decisions

Extract 53:

SD A: I've got a question for SD B, which was where does the - I found out the answer in the end actually - where is the applicant status. So you can search by everybody who is bank approved or limited bank.

HR A: I think realistically I don't think they will get back film on this data. It will be one of those things, as we move forward we will start putting

them in. Or if you run a search and you find a person, you will add their form and data.

SD A: Which file would have employees for which working pattern?

HR A: Oh, that was weeks ago, uhm! - it would have been called something like – is there one called employee working pattern or...

SD A: Yea there is.

HR A: That's the demo one. I am sure I made one...

SD A: Yea that's another thing, NetSuite is doing an update on our system tonight, so somethings like the L&D external provider the ways that's managed will be affected by that, so I didn't put that in yesterday, know that...

(Date 4.4.13)

The extract focuses on working patterns, which are different times and shifts of employees. These are made into samples and help payroll calculate the hours of an employee and their payment. HR A has previously worked on a working pattern sample to feed the system; however, it was some time ago. During this discussion SD A tries to remember the name of the file for HR A and SD A to work on it further. During this process HR A decides that the working patterns will be fed into the system as they go due to time constraints. Within the NDM model Figure 23 shows the process.

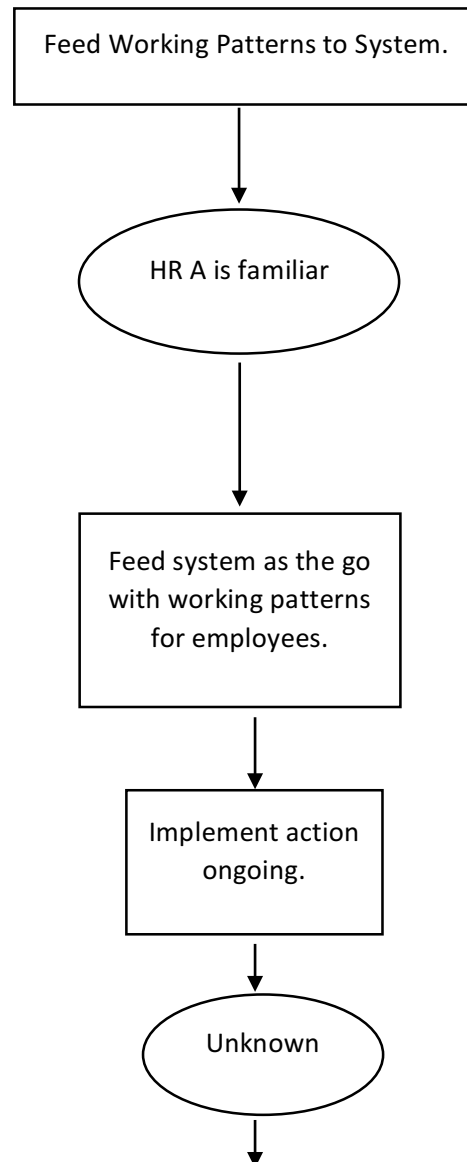


Figure 23 - Extract 53 Recall Decisions

5.4 Summary

This chapter demonstrated the link between tacit knowledge and the collected data. Within this chapter, it was shown when tacit knowledge surfaced during the meetings. Subjects were discussed over time and previously gained knowledge was used to solve problems within the project. It highlighted the various elements of tacit knowledge which can surface during a meeting such as socialization, extermination, internalization,

combination or constructive learning. Utilizing these elements, project members could move forward in the project and share their knowledge with the group. The selected extracts help show the occurrence of each element and when they surface. The extracts were combined within their fields to help analyse which topics were the most important and challenging throughout the project. Building on each other's knowledge, and their respective fields of expertise, decisions were made by the project member most familiar with the situation. This aided each project member to gain knowledge from the others and share it through different ways.

Having demonstrated the various ways tacit knowledge was evident throughout the project, and having previously discussed the literature linked to the data, they now need to be put into context to create the model. The model aims to determine how tacit knowledge influenced the individuals as well as the group. Decisions could be made with more merit by building on each other's expertise. The constant exchange in the dynamic environment allowed tacit knowledge to spiral and be exchanged within the team. This chapter presented the conversations and the elements found within. These elements will be put into context and used to build a model representing the interplay between individual and group tacit knowledge.

Chapter 6: Derivation of the Framework

6.1 Introduction

In the previous chapters, we have demonstrated how tacit knowledge flows in a software development project over time. Within each research cycle, different aspects of knowledge surfaced and were used to further develop theories. The main tacit categories were defined and how they were obtained was explained. The previously discussed research cycles and their results were analysed in accordance with the data in the previous chapter. The aim now is to examine the results in the context of the models in the literature, and construct a model with the newly found categories of tacit knowledge.

The assessment of the conversations supported the surfacing of tacit knowledge through different channels. Using the main criteria of Nonaka and Teece (2001), socialization, externalization, internalization and combination were established. In addition, group tacit knowledge and constructive learning were analysed. Throughout the process, different triggers were found, which allowed project members to create and share individual tacit knowledge. This process enables team tacit knowledge to flourish and therefore the project to advance.

To begin, the analysis of tacit knowledge triggers will be discussed. Each trigger - visual, conversational, constructive learning, anticipation and recall – will be analysed in reference to the extracts. The model will then be built with the data and theories acquired through the previous chapters. Finally, the model will be presented and discussed, followed by a summary of the chapter.

6.2 Tacit Knowledge Triggers

The most significant finding of the research cycles was the relationship between tacit knowledge triggers and knowledge exchange. These triggers as previously discussed emerged from the data but are based on the ideas of Clarke (2010). This section focuses on the analysis of tacit knowledge triggers data. The 45 extracts, shown in the previous chapter, were used to establish tacit knowledge in the project, providing a basis for further investigations of triggers found within each conversation. The different triggers are related and intertwined at times, which allows knowledge to trigger through different channels. In the previous chapter, the existence of triggers was established; however, no further investigation was made. This section is structured around the different triggers found throughout the extracts presented in the previous chapter; it provides an extensive investigation of the found triggers. Later, the established triggers are used to build the model identifying individual and group tacit knowledge within a software development project and when triggers commence within the process. Below, each trigger is discussed in relation to the analysed data extracts of the previous chapter.

The five main categories of triggers are:

1. Visual Triggers:

Tacit knowledge surfacing through visual stimuli.

Looking at previous notes or looking at the software enabled tacit knowledge to surface.

2. Conversational Triggers:

Tacit knowledge surfaces through a conversation held within the team.

These are very frequent, here tacit knowledge surfaces while discussing topics related to the project.

3. Constructive Learning Triggers:

Tacit knowledge is enabled through a team member explaining and the others learning from them. The trigger is within the person learning from the explicit knowledge.

Constructive learning triggers surfaced regularly when the software development team explained the new software to others in the team.

4. Anticipation Triggers:

Tacit knowledge was exchanged by an individual in the group by waiting for the topic to come up or the meeting to take place.

5. Recall Triggers:

Tacit knowledge resurfaces through discussions or visual aids which seemed forgotten or not present by an individual.

6.2.1 Visual Triggers

Visual triggers allow an individual to utilize previously gained knowledge to surface by reading or seeing information. During the research this trigger mainly surfaced when the software was looked at and edited by the team. The knowledge is gained tacitly, processed, which then triggers a socialization within the group. In these scenarios, the software development organization would present the developed pages to the human resource consultancy. The pages in the software were analysed by the team and changed according to their needs when possible. This mainly focused on wording, the layout or process in which the pages were to be found and structured within the software.

Visual triggers were found on numerous occasions such as during extract 6 where SD A explained the pensions pages. Through constructive learning, the HR consultants learned how the pensions pages functioned - during the explanations, HR A stops the lecture for a previously seen page. Although, SD A had moved on, HR A was still processing the visually gained knowledge in the previous page and asked to go back to see if a feature was available. In extract 11 one specific part of a page triggered a conversation within the group, the payroll ID. The work reference and the ID were confused by SD A, thinking two references were used by the HR organization; this triggered HR B to further ask and explained the referencing system, in which employees are categorized. This visual trigger allowed

conversational triggers to surface by starting socialization between the project members.

Visual triggers can also be more simplistic in their essence. In extract 12 the team looks at the salary screen, and needs to rearrange the order to fit the requirements of the HR consultants. The visual stimuli of the software triggers work and process knowledge of the HR team to be combined with the software engineering environment. A similar situation can be found in extract 17, where the 360 feedback is being assessed. HR A says changes within the structure of the pages will need to be done to fit the requirements of the client. HR A's tacit knowledge base of the customer as well as experience are combined with the knowledge visually gained through the software.

Throughout the data analysis there have been several extracts demonstrating how visual mediums trigger knowledge within an individual. This triggered knowledge enables the project team to further conversation and complete gaps of knowledge within the group and allows group tacit knowledge to prosper. Visual triggers launch an internal process within an individual, where the tacit knowledge base is used to combine the current tacit knowledge of an individual with the new visually gained knowledge.

6.2.2 Conversational Triggers

Conversational triggers occur frequently during meetings. Knowledge surfaces explicitly, which is then processed by a team member. The individual will then use the newly gained knowledge, add it to their existing knowledge and create new tacit knowledge. Socialization continues within the group and allows knowledge gaps to be completed. Due to conversations being at the centre of the research, conversational triggers are one of the most frequent mechanisms which are found throughout the research.

The very first extract analysed demonstrated a conversational trigger, where HR A discusses the pay policies, this then triggers SD A's tacit knowledge, where the topic is changed to payroll. SD A listens to HR A and

CL A discussing a finance related topic enables the recall of an unsolved issue with payroll. Later in the discussion, which seen in extract 2 HR A furthers the topic of payroll by building on the knowledge SD A shared. Through explicit exchange within the group, knowledge spirals and builds individual knowledge within each individual. Topics of discussion are altered and enhanced by using the tacit knowledge gained from the previous group member. Their similarities trigger socialization and externalization such as the conversation in extract 7, where the discussion allows knowledge to spiral and prosper within the group. Externalized knowledge is used by several members of the project, processed and complemented by the knowledge of each individual taking part in the discussion.

In extract 19 visual, conversational and constructive learning triggers interplay. While the software pages are being shown, conversations are being triggered and furthered within the group. This also allows constructive learning to take place. This combination can be found in several extracts such as 22, 29 or 31.

Conversational triggers are explicit communication within the group allowing group tacit knowledge to build. Each individual can utilize the knowledge to their advantage and complete missing pieces of their work to achieve project success. This trigger is often evident in combination with visual or constructive learning, where an external verbal medium allows an individual to take in information, process it, and reflect the knowledge to then externalize the new processed knowledge. This greatly supports group tacit knowledge and the core of a meeting, “to get everyone on the same page”.

6.2.3 Constructive Learning Triggers

A constructive learning trigger occurs when a project member explains to the others a specific topic of the project. It is a specific learning encounter rather than a conversation. The knowledge is passed on from a person explicitly to the group, which tacitly utilizes and combines the knowledge. During the project, learning was crucial due to the software being tailored to the organization. Each project group, the HR consultants, software

developers as well as the customer, exchanged knowledge through learning and integrating the knowledge in the software as well as its usage. This trigger also results in socialization, where questions are raised to clarify and add to the subject.

When SD A explains the pay by period page to the HR consultants, constructive learning takes place. This allowed HR A to process the gained knowledge and externalize what had not yet been understood. Externalization of knowledge can also be to confirm newly gained knowledge, such as in extract 5. SD A explains payments, which then triggers HR A to confirm the name of annual basic pay, FTM.

Constructive learning can also be task related. During extract 13, the customer as well as the HR team are trying to understand what data can be fed into the system and how it should be structured. This allows an interplay between constructive learning and conversational triggers, which can also be found in extract 16, where knowledge surfaces by teaching as well as learning and ultimately understand an issue of the project. Conversational triggers can also often be triggered by visual triggers. In extract 32, at the recruitment page in the software, a conversation is triggered on how the employees are ordered, by usage or alphabetically. Here, the visually, explicitly gained knowledge triggers a thought process within each individual, which is then turned into a conversation where knowledge surfaces through discussion.

6.2.4 Anticipation Triggers

An anticipation trigger allows an individual to raise a topic within the group, which he or she had waited to address. The trigger surfaces through a similar topic of discussion and allows a change of topic. In this case, the project member plans to talk about a subject during the meeting, however waits for a moment to bring it up. This is not to be put in direct comparison to a 'to-do-list' or minutes, where the subjects of discussion are being listed before a meeting and discussed one after the other, but rather allows

another to emerge through its similarity. It can surface during externalization or socialization.

During extract 1 SR A was anticipating discussion of payroll during the meeting, and although a conversational trigger allowed the finance topic to emerge, SR A was waiting to share payroll knowledge. Another example of an anticipation trigger is demonstrated in extract 18 where HR A asks to run through the 360 feedback. Here an email was sent to the group about the topic. It was not necessarily planned to discuss the topic; however HR A specifically asks CL A to explain and run through the process. This built on the previous meeting between SD A and HR A found in extract 17.

Anticipation triggers are the least commonly found triggers within the data. The meetings were usually structured around a specific topic of the software which was addressed. Unlike recall triggers, where knowledge pops up, anticipation triggers build around the notion of waiting to discuss a topic when the meeting allows the subject to come up.

6.2.5 Recall Triggers

Recall triggers surface when a topic of discussion or a visual trigger allows an individual to remember knowledge related to the subject which seemed forgotten or not shared in its entirety. This trigger can occur during any stage of the tacit knowledge process. New gained knowledge is processed through several steps, when it is initially heard or seen, combined with existing knowledge or when it is transformed into explicit knowledge and shared with the group recall triggers can emerge. This can change previously shared knowledge and alter the conversation. These triggers have a great meaning due to the knowledge almost being forgotten as well as the knowledge being at risk of not being shared in its entirety or differently could change the outcome of parts of the project.

During extract 4 SD A explains the monthly allowance page to the HR consultants, and during this discussion HR A asks how allowances are authorized. SD A first replies quickly, but then goes into more detail when

recalling that the short answer was not sufficient to understand the authorization process. This internal process allowed SD A to clarify and further the discussion. Recall triggers can also be minimal, where an individual mistakes one thing for another. Another example where a project member recalls previously gained knowledge from the project is in demonstrated in extract 37. HR A recalls a conversation from the day before and combines the current topic, processes and the previously gained knowledge to fill in gaps of knowledge.

In extract 5, HR A confuses FTA with FTM, which is a tacit process where, through knowledge recall, the initial thought is corrected. In extract 24, HR B recalls previously gained work knowledge and shares it with the project members. The conversation focuses on recruitment, where HR C is the recruitment expert within the group. HR B's knowledge is triggered through HR C's uncertainties and is able to add valuable knowledge, having previously worked in the field.

Recall triggers are quite frequent throughout the meeting. They are often found in combination with conversations, constructive learning and visual stimuli. Recall triggers are an internal tacit process where knowledge 'pops up' at random. This might be related as well as unrelated to the discussed topic. This trigger allows an individual to communicate knowledge which is recalled in order to further the knowledge exchange within the group and its group tacit knowledge.

Project success is allowing a group to constructively exchange knowledge and letting the other individuals in the group to learn the expertise of a member. This can be done through externalization, socialization or internalization. The interaction between individual and group tacit knowledge being essential not only sparks individual and group tacit knowledge but can lead to knowledge triggers and decision making processes. The following section demonstrates the interplay between socialization, externalization and internalization. It also takes constructive learning into account and how social interaction can lead to constructive learning and vice versa.

6.3 The Model – Recognizing and Harnessing Tacit Knowledge

In the first section of this chapter, the basis of the model was built by establishing tacit knowledge within the data and what kind of tacit knowledge was found. The interactions between parties and in what ways they communicate helped to create a basis for model development. Looking at tacit knowledge and when it surfaces within a group, as well as an individual, provided the basis for the development of the model. A detailed view of knowledge in- and output (constructive learning, social interaction, internalization or externalization etc.) identified at which point tacit knowledge surfaced within the conversations. These corner stones lead to an extended view of internal knowledge and group tacit knowledge which is enabled through several triggers within the dynamic environment. Furthermore, the first section of this chapter demonstrated triggers which allowed tacit knowledge to surface during the meetings. These triggers allow a more in depth view of what allows tacit knowledge to be exchanged during a meeting as well as the medium it is communicated with. Using the decision-making process shown at the end of the previous chapter establishes the usage of the knowledge and its influence within the group. This data is harnessed to support the development of a model and to graphically demonstrate the interrelations between the main theories used to analyse them.

One purpose of the model is to demonstrate that a meeting is a dynamic environment for knowledge exchange. It is a place where expert knowledge is combined and further developed. “Individuals draw from the team tacit knowledge and create their own tacit knowledge, which is fed back to the team. This is a background process which is dynamic and reciprocal relying on constructivist situated learning” (Ryan, 2013).

6.3.1 Combining the Models in a Meeting Context

In the previous sections, the existence of tacit knowledge in a software development project within a meeting context was established.

During the data analysis, the question as to how tacit and explicit knowledge are being exchanged in a software development project was addressed. The evidence presented was directly linked to the theories of Nonaka and Teece (2001), Ryan (2013) and Clarke (2010). The various elements of each theory - mainly focusing on socialization, externalization, internalization, combination, group tacit knowledge and constructive learning - were presented. In addition to these theories, Clarke's (2010) tacit knowledge spectrum introduced the concept of a trigger which was further investigated and differentiated into categories of visual-, conversational-, constructive learning-, and recall triggers. Utilizing models presented in respect to tacit knowledge by each researcher and combining these with the evidence found in the data is the focus of this section.

The aim of the model is to show the interplay between project members from an individual and group tacit knowledge perspective. In addition, the model helps develop an understanding of how tacit knowledge spirals within a software development team and supports the importance of verbal communication through meetings within software development project teams. Using all the assets, the developers, customers and experts of the subject regarding the software, aids in fully understanding, and the building of, knowledge within the group.

Using the three research cycles, the model was developed over time. The theories of Nonaka and Teece (2001) are used as the basis of the model, due to them developing an environment of knowledge exchange. Nonaka and Teece's theories were used in all research cycles. Adding Ryan's (2013) group tacit knowledge and Clarke's (2010) tacit knowledge trigger approach makes the model more detailed regarding individual and group tacit knowledge. Ryan's and Clarke's theories surfaced in the second research cycle. Finally, during the third cycle, the primary research findings were found, such as the tacit knowledge triggers.

Nonaka and Teece (2001) establish the environment for tacit knowledge to be created and exchanged within the 'Ba' environment, seen in Figure 24. This is the basis of the model and, within the context of

investigated data, represents the space of the meeting. The environment can be created through a conference call or face-to-face conversation. The dynamic environment where tacit knowledge can be exchanged is established and allows knowledge to spiral within the group through knowledge exchange. Using the theories of Nonaka and Teece (2001) as the basis of the model helps build up the group and individual tacit knowledge exchange.

Ba Environment

Figure 24 - 'Ba' (Nonaka and Teece, 2001)

Nonaka's SECI is then utilized to analyse the data and establish when socialization, externalization and internalization has taken place. The combination concept was excluded in the model due to the focus being the meetings which mainly focus on verbal communication and/or visual factors influencing the conversation. The SEI (rather than SECI) demonstrates the movement of knowledge, and its continuous connection between the quadrants. The Ba environment is represented as the backdrop, with the three means of exchanging and creating tacit knowledge spiralling within, seen in Figure 25.

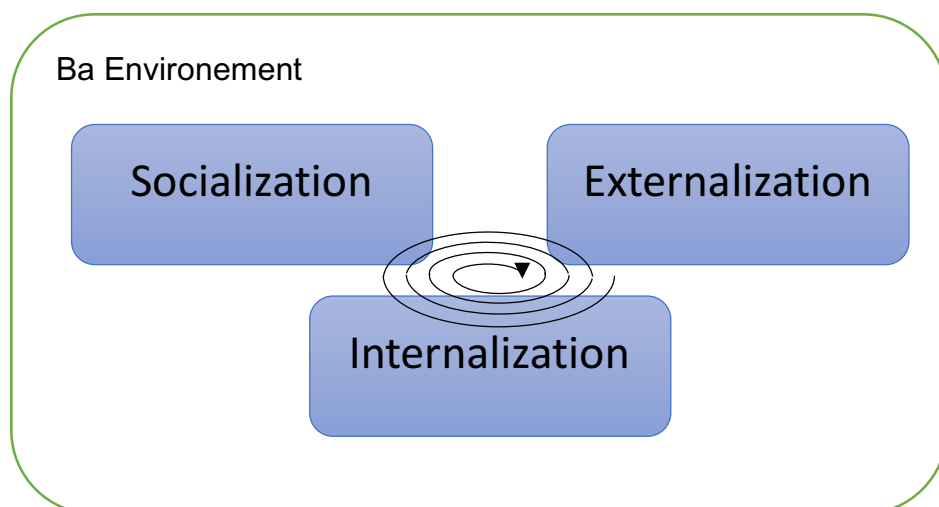


Figure 25 - 'Ba' and SEI (Nonaka and Teece, 2001)

The next building block of the model is Ryan's (2013) theory of the TTKM (Figure 26), which demonstrates group tacit knowledge and its integration within the transactive memory. This shows, in more detail than Nonaka, the process of tacit knowledge within a group, but overlaps can be found between the two theories. When combining the two, similar processes can be found which are compared in the Table below. Nonaka's socialization is represented by Ryan through the knowledge acquired and shared through social interaction, externalization is tacit knowledge acquired through constructive learning and finally internalization is individual knowledge and the transactive memory of the group. Team tacit knowledge plays a vital role within the research and will be added as a theory to the model. Nonaka and Clarke do not focus on team tacit knowledge within their research, but Ryan gives the insight on group tacit knowledge to the data. Other human factors were not in evidence within the data, and will therefore not be used within the model.

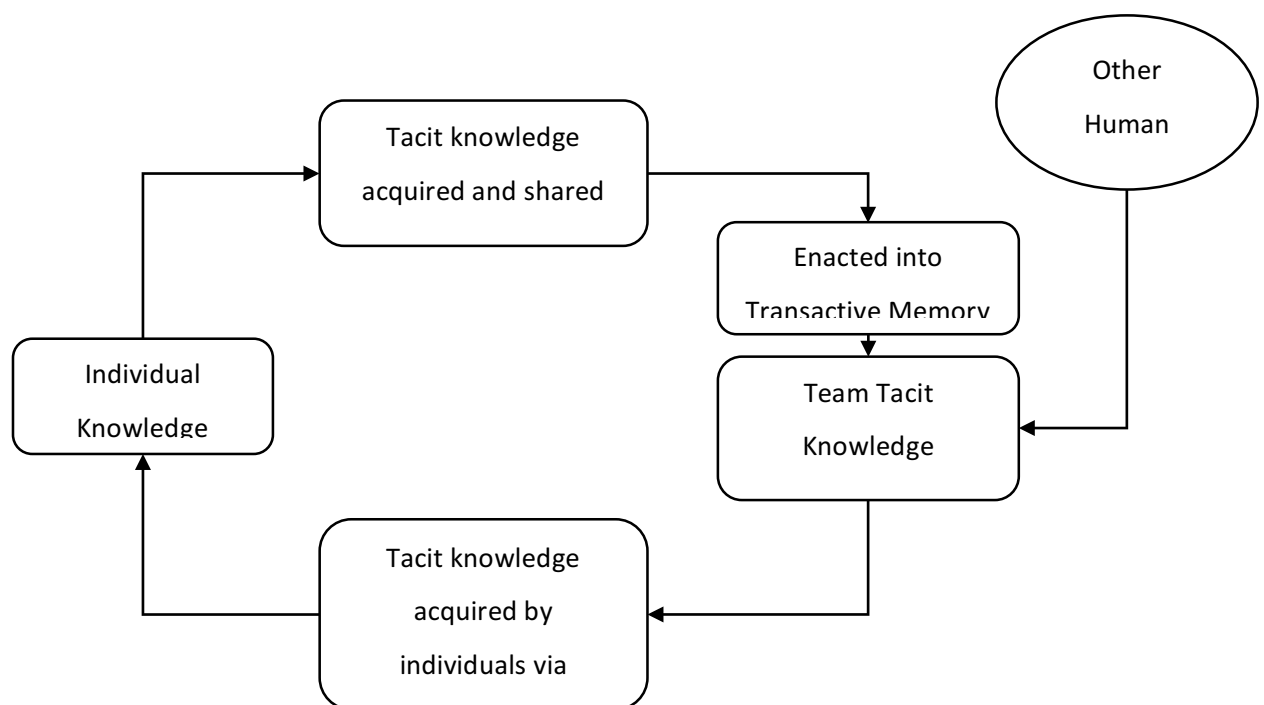


Figure 26 - Theoretical Model for the Acquisition and Sharing of Tacit

Adding these elements to the model demonstrates the ‘Ba’ environment, with the three exchange blocks spiralling within. This establishes the continuous knowledge creation, through different stages of knowledge. It is exchanged, created and internalized. Socialization indicates social interaction, internalization, the process of making the knowledge one’s own and finally externalization the acquired knowledge through learning. The theories of Nonaka and Ryan are compared in Table 97. Figure 27 illustrates Nonaka and Ryan’s theories and their similarities.

Table 97 – Nonaka and Ryan comparison

Nonaka	Ryan
Socialization – tacit to tacit	Tacit knowledge acquired and shared through social interaction.
Externalization – tacit to explicit	Tacit knowledge acquired by individuals through constructive learning.
Internalization – explicit to tacit	Individual knowledge / Enacted into transactive memory.

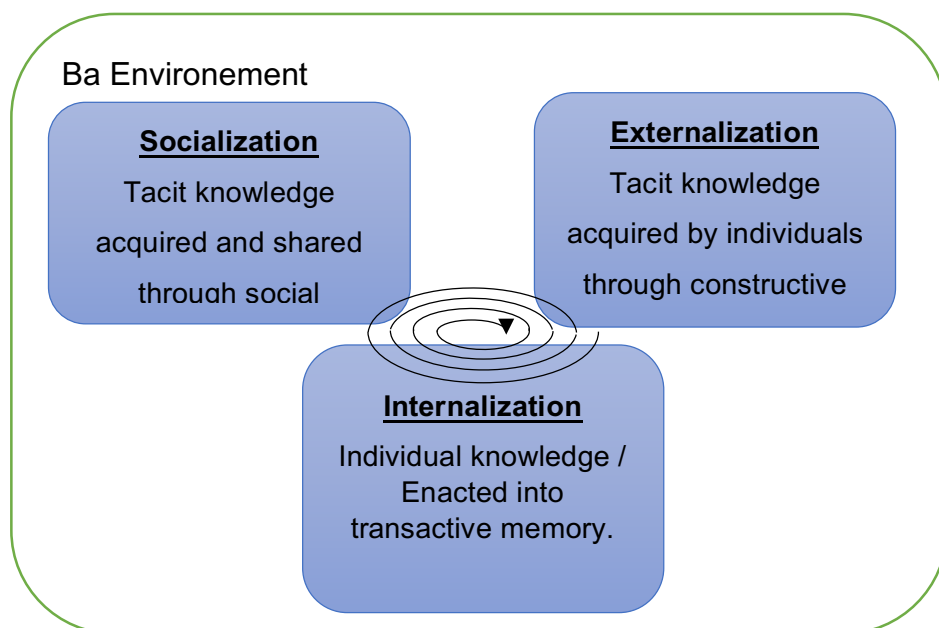


Figure 27 - Combination of Theories Nonaka and Teece (2010) and Ryan (2013)

The last main theory used to construct the model is Clarke's (2010) tacit knowledge spectrum (Figure 28). It focuses on the individual's tacit knowledge acquisition and processing. Clarke demonstrates the reflection process and how tacit knowledge is internally processed within an individual. Knowledge input commences the process; different stages of knowledge intake make the knowledge individual knowledge. He is also focuses on tacit knowledge triggers; however he does not categorize them and does not put them into a greater perspective in relation to the group. In the Table below the three different theories are put into comparison. Clarke's knowledge in- and out-put can be found in Nonaka's socialization and externalization as well as in Ryan's social interaction and constructive learning. The three theories are compared in Table 98. Finally, the models of Ryan and Nonaka, internalization, individual knowledge and enacted transactive memory stages, indicate the process in less detail whereas Clarke shows the entire internal process through reflection, its triggers, tacit elements as well as existing knowledge.

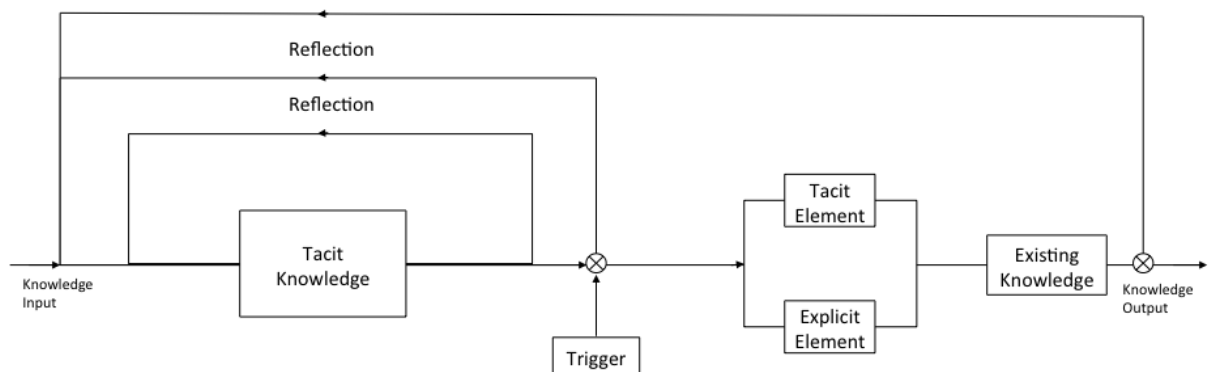


Figure 28 - Tacit Knowledge Spectrum (Clarke, 2010)

Table 98 - Nonaka, Ryan, Clarke Comparison

Nonaka	Ryan	Clarke
Socialization – tacit to tacit (face-to-face)	Tacit knowledge acquired and shared through social interaction.	Knowledge in- and output
Externalization – tacit to explicit (visual aids)	Tacit knowledge acquired by individuals through constructive learning.	Knowledge in- and output
Internalization – explicit to tacit (learning)	Individual knowledge / Enacted into transactive memory.	Process of acquiring and processing tacit knowledge (reflection – trigger – tacit and/or explicit element – existing knowledge)

Within his model, Clarke goes into great detail regarding the internal process, however within the data, not all elements could be validated or found. Focusing only on the existence of tacit knowledge within an individual, which has been established within the data as well as using newly gained tacit knowledge and processing, the tacit knowledge spectrum was reduced to just the process in the Figure below. Tacit knowledge, existing knowledge, reflection, triggers as well as the process of using the knowledge which can then be incorporated within the existing knowledge and reprocessing it will be the focus of the model. Figure 29 shows the process of internalization.

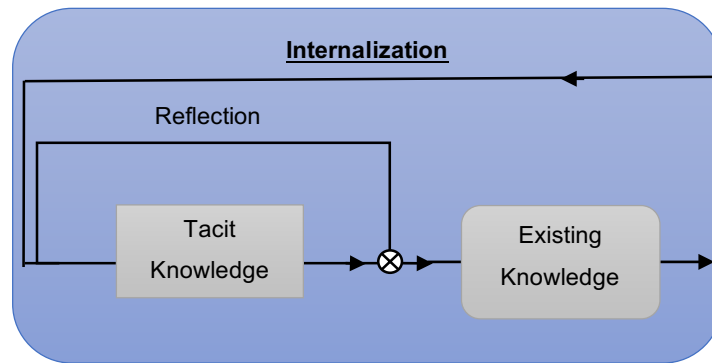


Figure 29 - Internalization Process

Combining the theories within the previously demonstrated models, Ba enables the dynamic environment to take place within the meeting, socialization and externalization are the keystones for the exchange of tacit knowledge and lastly, internalization allows each individual to process and take in the newly gained tacit knowledge, which can then be utilized to spiral the knowledge within the group seen in Figure 30.

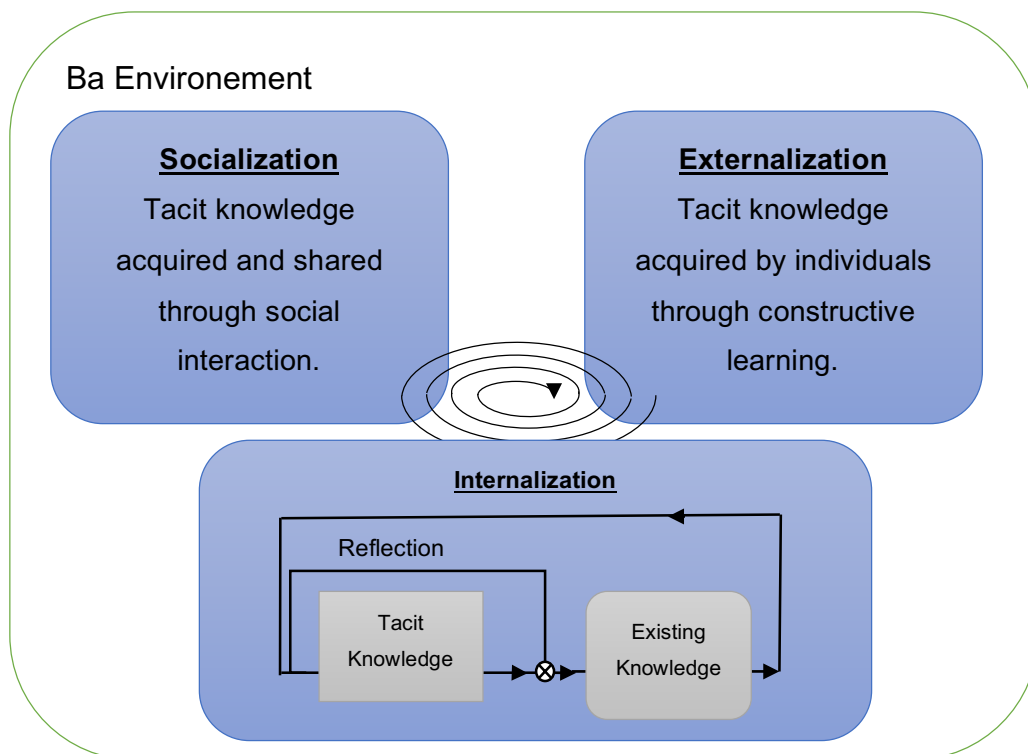


Figure 30 - Socialization, Externalization and Internalization

The model uses the three approaches from Nonaka, Clarke and Ryan to evaluate the data gained from the meetings. Throughout the data analysis not every aspect of each model could be validated, therefore they were reduced to their essentials and combined with the others in order to establish the four main areas of the model. Starting at the macro level, Nonaka's perspective, incorporating Ba, SEI and the spiral of knowledge, gives a basis for the meeting as a dynamic environment and the communication within. Ryan adds the group tacit knowledge perspective as well as constructive learning within teams. Finally, Clarke shows, at the micro level, the individual knowledge process as well as trigger points which allow tacit knowledge to surface within an individual. These theories are the basis for the construction of the model.

6.3.2 The Model and its relation to the exchange of tacit knowledge

In the previous sections, a framework for analysing tacit knowledge in software development projects has been established. This encompasses how tacit and explicit knowledge are exchanged within the meetings and identifies the different team's communication with each other. In addition, the evolution of tacit and explicit knowledge in a software development project and its impact on decision making was researched. The data was utilized to establish tacit knowledge triggers as well as showing evidence of tacit knowledge within the meetings, enabling theories to surface and validate themselves. Following on from this, the main theories and their relation to the models were discussed. This section will utilise the data presented in the previous chapter in conjunction with the model discussed above to analyse the data as well as the tacit knowledge triggers which surfaced during the research.

The internalization process found in Clarke (2010) has been found within the data. In this case, an individual takes in tacit knowledge communicated by another project member through constructive learning or socialization. The process of internalization is at the core of tacit knowledge,

since the combination of existing knowledge and new knowledge allows an individual to advance. This can then enable the group to have a common understanding of the project. Internalization can therefore be found in each extract of the data; however, some give more insight than others into the process of internalization.

The process of taking in tacit knowledge helps to understand the different elements of a project. In extract 11 SD A had wrong information on the payroll IDs, thinking there were two different numbers for the employees. Here the process of internalization was not made correctly, or the knowledge misunderstood. HR B then corrects and explains how the payroll ID's work. Hence, SD A had wrong tacit knowledge and needed to reflect and add to existing tacit knowledge with the new knowledge provided by HR B.

During extract 30 a knowledge exchange between SD A and HR C helps build the recruitment side of the software. In this case, SD A needs expert tacit knowledge from HR C confirming the processes put in place. While HR C internalizes the software visually, SD A receives feedback from HR C which is then processed internally. The exchange between the two project members allows the internalization processes to prosper. HR C will digitalize tacit processes, which need to be analysed and written down in order to have a process plan for the processes.

“What I am conscious of is that there are a lot of layers to do a simple thing. Now I take a phone call, I write it down. I pull up a word document, type it in and it's gone. I am literally going to be spending x amount of time, putting in all of this information, and remembering to go there, instead of it just being one page.”

Some of the processes seem more difficult when done digitally rather than writing them on a paper. HR C claims that some of the processes will tacitly be more difficult to do. The knowledge is taken in by each project member, processed and combined with existing knowledge. Upon reflection by each one solutions try to be found to better the software.

In extract 39, for example HR A reflects upon what SD A explained about the licensing. HR A takes in the knowledge through socialization and

uses the gained tacit knowledge internally. This process allows HR A to reflect upon the newly gained tacit knowledge and add it to existing knowledge.

“Interesting, so does that mean (uhm), so say if the HR consultant sets up as a buro and we would use 'chamber' as our client, does that mean no one at chamber could be an HR user?”

The raw tacit knowledge goes in where HR A is reflecting and trying to add it to existing knowledge. In order to understand and utilize the knowledge HR A responds with a question to validate assumptions. SD A responds to the question, during this time HR A reflects and through a continuous process uses the tacit knowledge and transforms it into existing knowledge.

Another example of the internalization process is found in extract 43, relating to project planning and training. HR C questions when the training should take place and whether deadlines are set right or whether they should be delayed. CL A explains through socialization the priorities of the customer to the group. These priorities are essential to the project team in order to adjust the work schedule and allow the software to evolve. The knowledge exchange allows the other group members to adjust and understand their individual priorities within the project, through the decision “to get it right” by the customer. This then allows SD A and HR A to adjust the project plan according to the new set deadline. The internalization process of each project member is to link their priorities to the new-found date.

“HR A: So, we can delay the payroll by a month as well and give us the 2 months we needed to get that ready.

SD A: This would also give us the time to see CL A and give her an advance notice of what is there. So that if she is raised any questions, she has an inside view of the system.”

These extracts show the influence of new knowledge on the project and how tacitly goals were altered and then exchanged within the group.

Internalization is a core component of tacit knowledge acquisition and utilization. When knowledge is not communicated fully or not understood

correctly it alters the project and creates more work. The time each individual uses to reflect and add tacit knowledge to existing knowledge can be very short but the process in itself is essential for project success. It is also the knowledge in-and out-put which enables group tacit knowledge to be created as well as socialization, externalization and constructive learning to take place. The model of Clarke is minimized to show the reflection process, new tacit knowledge and existing knowledge by an individual. A trigger point between tacit and existing knowledge is kept due to the tacit knowledge not being understood at first by the individual.

The next part of the model is the knowledge in-and out-put, which mainly includes socialization, externalization and constructive learning. This mainly focuses on the exchange between the project members before and after internalizing tacit and explicit knowledge. Looking at the software, learning the software, as well as discussions within the group are the basis of exchange. When taking in knowledge from a project member or seeing the software knowledge internalized an individual is responding to newly gained tacit knowledge. After the individual processes the knowledge is created and knowledge output occurs. The cycle spirals during the meeting as well as creating knowledge dynamically.

Knowledge in- and output is the essence of a meeting. This allows an individual to respond to gained knowledge by internalizing and processing it. In extract 14, HR A enquires about a specific access role within the software, payroll admin, which is knowledge input. One of the HR teams needs data within the software concerning payroll; however, this role should not have access to the full HR system where there is a broader view of each employee such as disciplinary record and grievances. SD A responds to HR A giving her the knowledge of the role of payroll admin. This knowledge is then internalized by HR A and used to further inquire about the role and what it can and cannot do, knowledge output. Here the internal knowledge of SD A is used by HR A to create tacit knowledge and further used and reflected upon, which then ends with knowledge output. This in turn triggers an internalization process of SD A. Once SD A has internalized the knowledge the cycle begins anew with knowledge output. Which allows a spiral to be

created, where knowledge is created through knowledge input, internalization and knowledge output. This process continues until the subject has completed or more information is needed to complete the knowledge cycle.

Another example of knowledge input and output is found in extract 22, where working patterns are the subject of discussion. Here knowledge from SD A, HR A and HR B is combined. Each individual uses the explicit knowledge transferred, knowledge input through socialization or externalization, and internalizes it, which then results into knowledge output. SD A explains how working patterns are built through constructive learning, HR A and HR B internalize the newly gained knowledge and combine it with their existing tacit knowledge and in return replies. Using the outline given HR A needs to complete the Excel sheet for the client so the live data can be fed into the system. In this case, the internalization process is linked to the combination of work done and work to be done by HR A. In addition, SD A still needs to acquire the correct formatting from SD B for the formatting of the Excel sheet in order for the data to be fed correctly. The subject is therefore completed once SD B has sent the correct formatting of the Excel sheet and no further question arise from HR A.

A final example of internalization is found in extract 42, relating to disciplinary records and grievances. HR A and SD B are discussing process and probation which are process steps within the system. SD B explains the process steps (knowledge output) and HR A then internalizes and responds with a question and once again creates knowledge output, which the next individual can then internalize through knowledge input. The cycle continues between the two until HR A and SD A have no further responses to the gained knowledge.

Knowledge in and output, in combination with internalization and a dynamic Ba environment, provide the basis for the knowledge transfer cycle and allow tacit knowledge to be created and used by the individuals within the group (Figure 31).

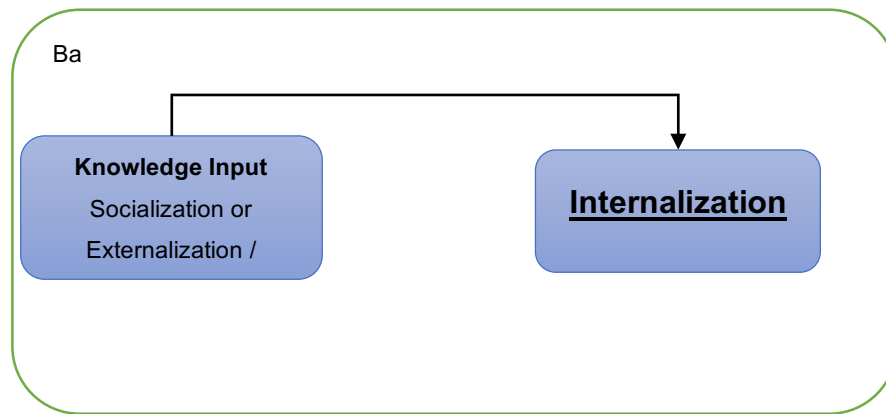


Figure 31 - Knowledge in and output to internalization

Focusing not only on individual knowledge, but also on group tacit knowledge, the integration within the model of a common understanding within the group plays a vital role. Once tacit knowledge has been communicated through socialization, externalization or constructive learning within the group, it is internalized and processed by each individual. Tacit knowledge is then added to existing knowledge by each project member, which allows a common understanding within the group of the newly gained tacit knowledge. Seen within the model of Ryan (2013), team tacit knowledge is established after the enacted transactive memory, where new knowledge is gained and stored. Unlike Ryan’s model, the process of internalization combines the transactive memory and individual knowledge by demonstrating the process of individual knowledge being taken in through reflect, tacit knowledge and existing knowledge. Once the process of internalization is complete team tacit knowledge is created, everyone within the group has a common understanding of the knowledge communicated. However, if an individual has not fully understood the shared knowledge, knowledge output allows individuals to add or ask for more knowledge from the group through socialization, externalization or constructive learning and the cycle begins anew within the dynamic environment.

Once the dynamic knowledge cycle commences through knowledge input by an individual in the group, the rest internalize it and respond when certain aspects are not fully discussed or questions arise. At this juncture, between internalization and knowledge output, a common understanding of

the discussed topic arises. Extract 15 demonstrates the discussion of additions and deductions, and in more detail, the annual or monthly payment of them. SD B asks how they are paid, where HR B responds which allows HR B to understand the current understanding of SD B and SD A on the subject. Through the response, SD A and SD B have a common understanding, team tacit knowledge, with HR B and HR A on the topic. Unlike information exchange, HR A and B are able to use the knowledge and create new tacit knowledge through internalization. This then results in knowledge output, where further knowledge from CL A is needed in order to complete the discussion of additions and deduction. Hence, the topic has not yet been fully discussed, but there is a common understanding of knowledge needed within the group.

Yet another example of group tacit knowledge is found in extract 25, where team tacit knowledge is created through the expertise of a team member. HR C explains through constructive learning the process of perusing a candidate for a job. SD A responds through questions, to gain a more complete picture of the process. This allows SD A and HR C to create team tacit knowledge of the subject by listening and responding to gaps transferred knowledge. Team tacit knowledge is created and understood by the present group, which allows the software to be completed with the knowledge transferred by HR C.

In extract 36, five team members are present to discuss the permissions and access to the software from different employees. HR B commences the discussion by asking which accesses CL A and CL B should have. This is quite important due to the payment of accesses within the software. Once again knowledge input commences the discussion through a question, CL D responds, launching the internalization processes and allowing team tacit knowledge to be created. This then results in knowledge output, a question, which then again CL D responds to completing the missing knowledge within the group, hence creating team tacit knowledge.

Team tacit knowledge allows the project team to understand and have the same level of knowledge. This is a key component of a meeting, where each individual gains knowledge as well as allowing other team members to understand knowledge sources and limitations from other project members. Once finalizing a meeting, the main goal is to have a common understanding within the team. The two internalization points are important to the model due to the interplay between two or more individuals. Each individual receives and processes the gained tacit and explicit knowledge from the project members, processes them and reuses them within the model, Figure 32.

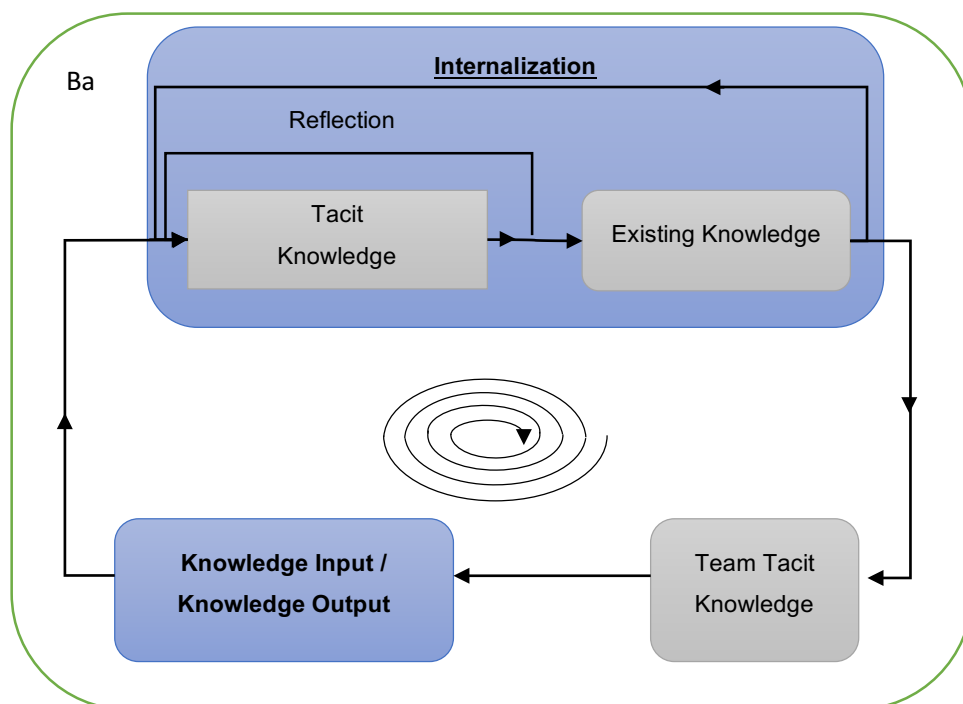


Figure 32 - Tacit Knowledge Model Process

Finally, the model adds trigger points at specific points within the model. These were found during the third research cycle. These are the main contributions of the data analysis in the development of the model. After having commenced the cycle with knowledge input – socialization, externalization or constructive learning – the internalization process commences. Within the process there are two trigger points, one at the very beginning and one in the middle. In addition, there is a third trigger point after team tacit knowledge, which then results into knowledge output. Due to the

knowledge exchange being a spiral, after knowledge output the internalization process within another project member begins anew with the same three trigger points.

Depending on the received tacit knowledge, the various trigger points are in different locations. Visual -, conversational -, constructive learning -, anticipation -, as well as recall triggers can all surface within different moments of the tacit knowledge cycle. These trigger points begin the tacit knowledge cycle and allow knowledge creation or knowledge sharing depending on the point where they are set.

The first trigger point is at the very beginning of the internalization process, this trigger point launches tacit knowledge internalization. The trigger point is often found in combination with explicit knowledge.

Within extract 3, SD A explains the difference between the payroll reports of the new and old software. At this point, HR D asks when the switch between the old and the new was to take place. HR D, as well as HR A, use the knowledge input SD A gives the team and combine it with their existing tacit knowledge, which in return allows team tacit knowledge to be created. During the spiralling knowledge exchange, each creates a knowledge output after internalizing knowledge from SD A, adding knowledge to the group and hence enabling team tacit knowledge to be created for the three-team member involved in the discussion.

Extract 12 allows team tacit knowledge to be created by adjusting a software page. The discussion with HR B and SD C concerns the salary page and the layout. Dates where pay begins should be seen. The trigger point allowing tacit knowledge to be processed and shared is at the beginning of the internalization process, HR B sees the layout, processes it and responds to the project group. This allows SD C to respond and agrees that no changes need to be made. Here, SD C takes in the explicit knowledge shared by HR B, internalizes it, and combines it with tacit knowledge previously gained from the project. Once HR B and SD C have agreed upon the changes the cycle of knowledge concerning this topic has ended at this point.

Tacit knowledge triggers can also surface in combination with others. An example of this can be seen in extract 43. Once again, the software is examined by the project group to learn the software as well as change features or functions which do not correspond to the needs of the client or the HR consultancy. SD B explains reviews and probations which are a process within the system, meaning it is a step-by-step process. This allows constructive learning to take place while visually assessing the system. HR A has two different trigger points which can allow tacit knowledge to surface. In addition, a conversation can be triggered by the group when an enquiry is made. SD B explains the process through knowledge output, allowing HR A to receive the knowledge through knowledge input. At the beginning of the internalization lies the visual trigger launching the process of tacit knowledge acquisition. In addition, SD B explains the page, making it more cohesive to HR A. HR A processes the gained knowledge and adds it to existing tacit knowledge. At this point a common understanding of the page is built between the project members. Through the reflection process a question, knowledge output, is triggered by HR A, allowing the cycle to begin anew with the person responding to the question, SD B. The question is internalized by SD B and other present project members, allowing a common team tacit knowledge to be created. When the subject is still not clear or changes need to be made, the cycle continues till the team is satisfied with the gained knowledge as well as the knowledge created on the page.

The decision-making process during the meetings was essential to enable the combination of tacit knowledge from each project member to advance within the project. During this time, knowledge was shared and combined by the person with the most expertise within a discussion. Allowing knowledge to flow between project members enables tacit knowledge to surface from each member of the project team. Sharing and exchanging the knowledge between each other is at the core of the model.

Looking at the employee centre in extract 52 demonstrates how tacit knowledge is used to make decisions and enrich the knowledge of the team. Learning constructively from SD B triggers a conversation between SD A, HR A and HR B. SD A remembers a niche within their employees, agency

workers, which are not often recruited but still need to be in a category. SD B triggers through constructive learning the internalization process of SD A. Recalling an earlier conversation, demonstrates an internal trigger which surfaced. Through reflection and combining tacit knowledge, the team is made aware of the problem. This then triggered a conversation about searches and how the tabs should be displayed. This extract portrays a discussion from several team members combining their knowledge and using it to each other's advantage. Levering on the exchanged knowledge from a previous team member, tacit knowledge is triggered. Allowing the knowledge to flow in a dynamic environment until no further questions or remarks are shared allows the team members to come to a common understand of the topic.

Within the model three different trigger points can be found after each internalization process. Each trigger type has a different moment of allowing the individual to seek tacit knowledge combination. Visual triggers are mainly found at the beginning of the internalization process. Through the tacit act of gaining new knowledge by seeing the reflection process is launched. Other triggers found at the beginning of the internalization process is the conversational and constructive learning triggers. Like the visual trigger, an external, explicit knowledge source launches the internalization process.

Within the internalization process another trigger is found, adopted from Clarke (2010), but has a different meaning. Clarke (2010) defines this trigger through group discussions, breakdowns or problems. In this model, the trigger is launched through an internal process. Here anticipation or recall triggers are mainly found. These are the most difficult to analyse due to them occurring within a person and while the tacit knowledge acquisition process is happening. Through the first trigger, this trigger is launched and used by the individual to combine and add to their existing knowledge.

Finally, the third trigger is found after team tacit knowledge. This trigger occurs once the new knowledge has been internalized and understood. This trigger allows an individual to respond to the newly gained knowledge. The final and third trigger happens when more knowledge is

needed for the topic. This trigger allows explicit knowledge to surface through conversation.

The model, shown in Figure 33, demonstrates how knowledge spirals and is passed on from one project team member to the next. Launching the cycle by an individual in the group allows the cycle to commence. This can be through constructive learning, conversations or visual influences. Knowledge is made available to an individual and processed, through reflection and combining the new tacit knowledge with existing knowledge. Triggers influence the internalization process, which can be at the beginning as well as during the process. Creating new team tacit knowledge, another trigger allows an individual to share knowledge output with the group. Once again this can be externalized through visual factors, conversations or constructive learning. This can then recommence the cycle anew beginning with the internalization process. The process will go on in the dynamic environment until the project group has no further questions, remarks or knowledge to share on the topic.

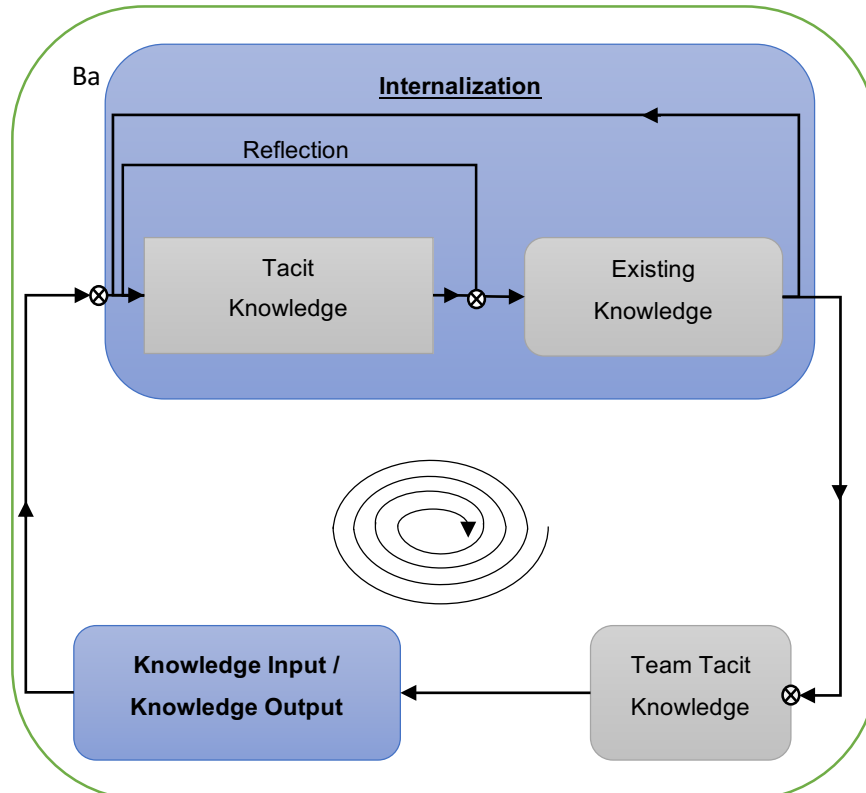


Figure 33 - Individual and Group Tacit Knowledge Model

Each research cycle added to the creation of the model. Beginning with the theories of Nonaka and Teece (2001), the data was analysed with a view to organizational tacit knowledge creation. Not knowing which aspect of tacit knowledge was to be investigated, different aspects of tacit knowledge emerged through the data. The focus of Ryan (2013) and Clarke (2010) surfaced through the second research cycle. Looking at knowledge creation and its flow within the group these three models allowed the analysis of the data. However, lacking the group and individual aspects as well as an understanding of what helped knowledge surface, required the construction of a new, refined model.

The assessment of the project and the group involved also played a vital role in the creation of the model. The three interest groups involved worked towards a common goal, but at the same time had different knowledge, needs and goals. These needed to be combined to make a usable and efficient product. The aim of the meetings was to help each organization add knowledge to the software, and exchange needed information efficiently and in a short time frame. The common goal was evident throughout the recordings as well as each organization's contribution. The purpose of the model is not to find barriers within the project team but rather to identify what supported knowledge exchange. These positive aspects need to be reinforced to help create a successful project.

The new model incorporates both group and individual perspectives. The interplay between team members and their importance is highlighted. The process of internalization is crucial for knowledge exchange. Using expert knowledge from each interest group and combining it with new knowledge gained from others allows the growth of knowledge in the group. Finding triggers, which allows others to add or amend knowledge exchanged in the group, helps project success. The model allows the reinforcement of those triggers in future projects as well as creating a space where knowledge can be exchanged.

6.4 Summary

This chapter focused on further analysis of the literature as well as the data itself. Showing how tacit knowledge was used during the meetings, and at which times it surfaced, was the specific aim of this chapter (as well as the thesis as a whole). The model was built on the gathered data, to identify the points of tacit knowledge creation and what triggers tacit knowledge to be created and then surface. The dynamic environment allows project members to exchange knowledge and utilize it for the benefit of the project.

Having demonstrated the occurrence of triggers within the project, through visual, conversational, constructive learning or anticipation as well as recall triggers showed how tacit knowledge can surface. Looking at triggers and when they allow an individual to use tacit knowledge and process it, demonstrates the importance external factors play in knowledge acquisition. Using the internalization process as means to combine and process the newly gained tacit knowledge, recall triggers allow a project member to remember what seemed lost knowledge. Through a common understanding of the topic at hand, after the internalization process of the present project members, team tacit knowledge is created. This then can result into knowledge output, when uncertainties or comments still need to be made.

Spiralling knowledge allows a project team to grow, not only in their knowledge of the project but also in the expertise of each team member. Supporting each team member to share their knowledge with the group helps seemingly lost knowledge to resurface and to be shared. Face-to-face meetings, as well as conference calls, allow a vast amount of knowledge to be transferred in a short amount of time. Working together on a project helps speed up processes. The model helps advance our understanding of what triggers tacit knowledge to surface within a project, and thereby provides guidance for managers to create appropriate “spaces” within their projects for tacit knowledge creation.

Chapter 7: Results and Conclusion

7.1 Introduction

The research presented in the preceding chapters demonstrated how tacit and explicit knowledge is created and exchanged in software development projects. Shedding light on best practices to make the 'cannot be articulated' into 'fully articulated' has been a challenge throughout the research. The representation of tacit knowledge within a model which helps us to understand the importance of meetings and the knowledge exchanged within sets out to provide a basis for future research within the field.

Specifically, researching how to convert tacit knowledge into explicit knowledge can help focus strategies which can be adopted in modern, forward thinking organizations. Maybe one of the reasons why tacit knowledge is so difficult to transfer into explicit knowledge is because everyone learns and understands in his or her own unique way. Therefore, when exploring the challenges or difficulties posed by software projects, an individualized and customized approach may be needed. However, the applied model should help focus a project team on specific points where knowledge can be exchanged in the easiest manner that allows knowledge to surface from each individual in an efficient way. When utilizing the model, certain questions should be kept in mind. What makes some people learn faster than others? Will a collaborative approach work? What components should be in a training program for people using the software? What should the organization do as far as teaching people to teach others? What are the interpersonal competencies that are needed so that a project head can bring everyone on board and get the most out of one and all? These are areas of discussion and exploration that can productively be investigated within each project.

7.2 Key Conclusions

RQ1. What is the current understanding of knowledge exchange in software development projects?

As regard to the current understanding of knowledge exchange in IT software project development, a picture of current concepts and theories emerged. Exploring the literature helped build an understanding of what tacit knowledge is, and how it can be used to explore and better understand knowledge exchange and development within a case study. The theoretical outline was a necessary pre-requisite to the collection and exploration of the data during the project. The main concepts which surfaced for further investigation of the data included Nonaka and Teece's (2001) tacit knowledge spiral as well as their concept of the dynamic environment Ba; Ryan's (2013) team tacit knowledge measure as well as Clarke's (2010) tacit knowledge spectrum. These constituted the cornerstones of tacit knowledge investigation. In addition, the Naturalistic Decision Making (NDM) model was used to understand decision making in the project and was combined with the previously mentioned tacit knowledge concepts of analysis. Furthering this investigation helped combine the knowledge gained over time and its impact on decision-making within the project team.

RQ2. How can tacit and explicit knowledge be recognised and evaluated in software development projects?

Building on the concepts and theories of tacit and explicit knowledge, the data evaluation demonstrated how tacit and explicit knowledge were exchanged. In more detail, the focus was on the moments when tacit knowledge surfaced as well as its transfer into explicit knowledge. In addition, the research explored when the knowledge was received by the group from an individual and the dynamic response which came by receiving and processing tacit knowledge. The data analysis helped the utilization of the gained knowledge and allowed a further investigation into its connections within the group.

RQ3. To what extent does non-communicated tacit and explicit knowledge amongst team members influence the project and its acceptance?

Exploring the evolution of tacit knowledge within software development projects and its effect on individual and group decision making was the next objective of the research. Focusing on the NDM model in combination with the previously explored data, the decision-making process was evaluated. This helped understand when tacit knowledge surfaced in order to further the project and decide how to move forward. These decisions were at times made by an individual as well as by the group, depending on the situation as well as what was needed. However, the decisions made were often directly transferred to the software as well as to the project as a whole. Here, the knowledge transfer was vital in order to make appropriate decisions weighing in the needs and capabilities of each project member. Another part of this investigation were previously made decisions, recall decisions, where a subject was previously discussed or agreed upon, however the previously transferred knowledge at times forgotten or not used instantly. These recall decisions enriched the interplay between group and individual tacit knowledge over time, as well as its importance for regular and clear communication within the team.

RQ4. Can tacit and explicit knowledge be better harnessed through the development of a conceptual model for use in software development projects?

Finally, the model was created by combining the theories and data collected. This addressed the last question of the research – whether it is possible to recognize and harness tacit knowledge through a conceptual model within a software development project. Models and concepts found in the extant literature helped build the model. Combining these theories helped tacit knowledge to surface within the data and demonstrated the exchange within. In addition, using the decision-making process highlighted the

dimension of time where tacit knowledge is exchanged and its relation to the group and individual knowledge transfers. Identifying when tacit knowledge surfaced, and at what moments during a discussion knowledge was triggered within the team, created a more detailed view of tacit knowledge transfer during a meeting.

Throughout the research, the constant aim was to further understand and progress the field of tacit knowledge transfer within a software development environment. Using current theories and methodologies to explore data and transferring it within a model will help project teams focus on exchanging and exploring knowledge from different sides. Tools such as constructive learning within the group as well as discussions to further understand the software and exploring the knowledge input from each individual is crucial for a project to succeed. Trigger points, which support the transfer of tacit knowledge into explicit knowledge within a group, were created. However, these can only be created within a dynamic environment in which an exchange of knowledge is supported by the project team. Spending time together as a team and working together is at the core of knowledge creation and transfer. Bouncing ideas off one another and subsequent mutual learning furthers the knowledge creation process. This allows each individual to take in more knowledge and provide a better, more complete view of the subject and enables the prospect of a more complete software to emerge.

7.3 Methodological Issues and Limitations of the Research

Progressing research in this area is not expected to be easy because the topic area is not tangible and needs qualitative in-depth research in order to reproduce and understand the project studied, as well as the people involved. The mind is what is being discussed and researched, and understanding and evaluating shared knowledge by each individual is challenging. A learning environment where each project member can benefit from the knowledge of the group is the ideal environment for tacit knowledge

to surface. A customizable, individualized, dynamic learning environment is the way to go, which can be created through meetings and having a qualified and resourceful project team. However, how should all the pieces fit together? Is it necessary to be in a conscious learning environment, or does it just come naturally? Are there certain tools that might facilitate transforming tacit knowledge into explicit knowledge? The more one studies the literature, analyses and evaluates cases, the more it is plain that no one seems to have a truly comprehensive or prescriptive template for what the perfect learning environment will be. Tacit knowledge is somewhat of an area of mystery, and turning what we cannot even articulate at times, into reproducible and effective explicit knowledge and allowing others to memorize and internalize this knowledge is a great challenge. That being said, building blocks and theories help analyse such situations and allow us to have a glimpse at exchanged tacit and explicit knowledge within a group. Keeping in mind that each individual has more knowledge to share and utilize during a project, each situation will need specific tailoring to understand what triggers knowledge exchange. Nonetheless, finding a formula for a software development projects and its tacit knowledge transfers is a fundamental step in ensuring that companies can meet the market demands of complex software projects.

In addition, another important subject to be explored is managing intellectual capital and organizational knowledge in such a fashion that interests are not lost. An organization that wants to encourage learning and constant growth must also be aware of the rapid loss of knowledge once a project has been completed and parts of the project team have left. Communication internally must be free-flowing and constant during a software development project; however, once the project has gone operational the communication is often fades to a minimum. This however does not mean that it does not need constant improvement. Making sure that the lessons learned during the project by the group is not lost, as well as key players needing to stay in contact in order to ensure the successful implementation of the software and overcoming the first challenges during the operational stages, is crucial after the completion of the project.

Similarly, a literature review into the subject area can reveal and help understand managerial competencies. Is there something about the organization's management team that might be worth learning? Are they transformational leaders? And what is the role of transformational leadership in creating a learning culture where tacit knowledge becomes explicit knowledge and back again? Workers are commonly confronted with their own deficiencies when identifying why a software development project falls short of expectations. Yet, a progression of the research should show what competencies are needed by senior management types, so that they can rationalize their organization in a manner that creates a well-functioning project team. The competencies of such a team are vital for efficient transfer and exchange of tacit knowledge. Without diving too deeply into the available literature, the importance of transformational leadership, working together as a team, creating a dynamic environment for knowledge exchange, persistent modelling, shortened feedback loops, and the elimination of bureaucratic layers are all staples of good management in this area. Therefore, good management will include people who are open to the idea that a business place is also a classroom that treats employees as students. Senior managers, therefore, must be teachers in their own right and must pass on what teaching skills they can to subordinates. At the same time, it is crucial for the employees to be able to use each other's knowledge to further a project and for the managers to allow a team to work as an entity.

7.4 Recommendations and Future Research

Progressing the research, on a practical level, will not stop with capturing only the areas of inquiry illustrated above. In the future, the aim is to fully understand the group and individual levels of tacit knowledge within software development projects. Although, there is no perfect method to achieve this, it is important to further investigate tacit knowledge within a software development context. Further investigations could productively focus on knowledge transfer with companies where, through a qualitative approach, tacit knowledge can be analysed. Listening to conversations and

the exchanged tacit knowledge over time demonstrates the need for constant knowledge exchange as well as the importance of a well-functioning team. Understanding which factors allow tacit knowledge to surface and its utilization during the project should help future projects advance, and allow them to advance further in a more progressive way. It is hoped that a closer affinity with the subjects of the study will shed new light on how to confront issues that might frustrate efforts and hinder learning within the project – and this may thus ultimately enhance the likelihood of project success. Although the analysed project focused on software development, the model should be tested and used in other projects as well to see whether it can function in the same way.

7.5 Contribution and Implications

The aim of the research was to analyse the interplay of individual and group tacit knowledge in a software development project. Through a novelistic approach, the meetings were discussed and analysed to help demonstrate when and how tacit knowledge surfaced throughout the meetings. Using one project and analysing it intensely allowed the analysis of key players and how knowledge is transferred between the project members. Using the expertise from each project member, it could be seen which team member was the most essential during a topic. Highlighting the point of knowledge sharing and acquisition not only demonstrated tacit knowledge creation, but also how tacit knowledge from each team member is essential to create comprehensive tacit knowledge.

The creation of tacit knowledge and the usage of it by each team member was represented in a model demonstrating the spiral of tacit knowledge sharing and acquisition during a software development project. This model demonstrated the cycle of individual knowledge acquisition and at which points an individual is triggered to create new tacit knowledge. This is supported by the group and a common understanding between the different team members. Communicating knowledge within a team verbally

through face-to-face conversations as well as through conference calls allows the team to grow and understand the expertise of each team member more intensely. In addition, within the software development environment, it is easier to make amendments to the product due to it being virtual. Creating pages together and allowing the customer to contribute can lead to a more complete software product.

The model can help project managers create and plan knowledge exchange throughout their project. Looking at the model from a practical standpoint, a manager can use it to create dynamic environments for knowledge exchange and creation within the group. Knowing when trigger points may occur can help create and surface tacit knowledge, allowing the project manager to pay more attention when knowledge is exchanged and how. Not only having conversations over the phone, but also within the same room, allows project teams to speak more freely and use the exchanged tacit knowledge. After a project meeting the model can help develop an understanding of when instances of knowledge exchange are created within the team that helped further the project. These moments need to be recreated by the group, to not only allow individual knowledge to grow but also team knowledge.

The presented model should help researchers and practitioners further understand when tacit knowledge is exchanged, and by which means - through visual aids, conversation or constructive learning. The aim is to pay more attention during projects to when and how tacit knowledge can be extracted from a project member by using these aids. Allowing the others to respond to explicit knowledge shared by a project member helps further the knowledge within the group. A conversation is only then complete when the people involved have understood the needs and concerns of the other project members. As a project manager, a dynamic environment needs to be created where project members can exchange tacit knowledge and interact upon others. The aim of a meeting is to fill in gaps of knowledge within the project team and allow teams to work together. In addition, creating triggers for team members to surface tacit knowledge can help the creation and

sharing of tacit knowledge. Finally, it is essential to create 'Ba' through meetings where knowledge can be exchanged freely.

This study has demonstrated and developed several significant findings. The most important one concerns the trigger points, created by a group. Using the knowledge exchanged within a room and allowing individuals to use and complete the knowledge at specific points helps understand the flow of tacit knowledge. It has provided a connection between existing theories of tacit and explicit knowledge in the workplace, by building a bridge between individual and group tacit knowledge within a project and its importance. It has shown how tacit knowledge could be socialized in the knowledge exchange process for a software projects by building a model harnessing the knowledge exchange. It suggests a way in which teams can focus on this process for their mutual benefit by creating 'Ba' through a meeting where each project member is free to benefit from the knowledge and experience from others. It identifies the situated knowledge in these teams and suggests further work could identify how to establish the process within an organization.

References

A

- Agar, M. (1986). *Speaking of Ethnography*. Qualitative Research Methods Series, Vol. 2. Thousand Oaks, CA: Sage Publications.
- Agar, M. (1996). *The Professional Stranger: An Informal Introduction to Ethnography*, 2d ed. San Diego, CA: Academic Press.
- Agbim, K.C. & Oriarewo, G.O. (2012). Spirituality as correlate of entrepreneurship development. *Journal of Research in National Development*, 10(3), 154-164.
- Akgun, A.E., Byrne, J., Keskin, H., Lynn, G.S., & Imamoglu, S.Z. (2005). Knowledge Networks in New Product Development Projects: a Transactive Memory Perspective. *Information Management*, 42 (8), 1105–1120.
- Al-Ali, N. (2003). *Comprehensive Intellectual Capital Management: Step-by-Step*. New York, NY: Wiley & Sons.
- Al-Qdah, M. & d Salim, J. (2013). A Conceptual Framework for Managing Tacit Knowledge through ICT Perspective. *Procedia Technology* (11): 1188-1194.
- Argyris, C., & Schön, D. A. (1978). *Organizational Learning II*. Reading, MA: Addison-Wesley.

B

- Baldrige Glossary (2009). *Knowledge Assets* [Baldrige Glossary HD]. Retrieved from www.baldrige21.com.

- Banacu, C.S., Busu, C. & Nedelcu, A. C., (2013, November). Tacit Knowledge Management – Strategic Role in Disclosing the Intellectual Capital. Paper presented at 7th International Management Conference, Bucharest, Romania.
- Bentz, V. M., & Shapiro, J. J. (1998). *Mindful Enquiry in Social Research*. Thousand Oaks, CA: Sage.
- Berman, S.L., Down, J., & Hill, C.W.L., (2002). Tacit knowledge as a source of competitive advantage in the National Basketball Association. *Academy of Management Journal* 45 (1), 13–31.
- Bernard, H.R. (2011). Sampling III: Nonprobability Samples and Choosing Informants. In *Research Methods in Anthropology: Qualitative and Quantitative Approaches* 5th edition (pp.143- 155). Altamira Press.
- Bernard, H. R., & Ashton-Voyoucalos, S. (1976). Return Migration to Greece. *Journal of the Steward Anthropological Society*, 8 (1), 31-51.
- Berry, D. C., & Dienes, Z. (1993). *Implicit learning: Theoretical and Empirical Issues*. Hove, U.K.: Lawrence Erlbaum.
- Brooking, A. Board, P, & Jones, S. (1998). The Predictive Potential of Intellectual Capital. *International Journal of Technology Management*, 16 (1-3): 115-125.
- Boehm, B. W. (1991). *Software Risk Management: Principles and Practices*. IEEE Software, January 1991, 32-41.
- Boyd, C.O. (2001). Phenomenology the Method. In P.L. Munhall (Ed.), *Nursing Research: A Qualitative Perspective* 3rd. ed. (pp. 93-122). Sudbury, MA: Jones and Bartlett.
- Bozarth, Jane (2014). *Show Your Work*. San Francisco: Wiley.
- Busch, P. (2008). *Tacit knowledge in Organizational Learning*. Hershey, PA: IDEA Group.

C

- Cameron, M. E., Schaffer, M., & Hyeoun, P. (2001). Nursing students' experience of ethical problems and use of ethical decision-making models. *Nursing Ethics*, (8), 432-448.
- Chau, T., Maurer, F., & Melnik, G. (2003, June). Knowledge sharing: agile methods vs. Tayloristic methods. Paper presented at IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises. Linz, Austria.
- Clarke, T. (2010). The development of a tacit knowledge spectrum based on the interrelationships between tacit and explicit knowledge. Retrieved from:
<https://repository.cardiffmet.ac.uk/bitstream/handle/10369/909/T%20Clarke.pdf>.
- Clifford, J. (1997). *Routes: Travel and Translation in the Late Twentieth Century*. Cambridge: Harvard University Press.
- Cohen, W.M., & Levinthal, D.A. (1990). Absorptive capacity: a perspective on learning and innovation. *Administrative Science Quarterly*, 35 (1): 128-152.
- Collins, H. (2010). *Tacit and explicit knowledge*. Chicago: University of Chicago Press.
- Coy, M. (1989). *Introduction in Apprenticeship: From Theory to Method and Back Again*. Albany: State University of New York Press.
- Creswell, J. W. (1998). *Qualitative inquiry and research design: Choosing among five traditions*. Thousand Oaks, CA: Sage.
- Cross, R., & Baird, L. (2000). Technology is not enough: improving performance by building organizational memory. *Sloan Management Review*, Spring: 69-78.

D

- Daft, R.L. & Weick, K.E. (1984). Towards a model of organizations as interpretative systems. *Academy of Management Review*, 9 (2): 284-295.
- Denzin, N.K. (1970). The sociological interview. In *the Research Act: A Theoretical Introduction to Sociological Methods*. New Jersey, NJ: Transcation Publishers.
- Dewalt, K.M. & Dewalt B.R. & Wayland C.B. (2010). *Participant Observation*. Altamira Press; Second Edition.
- Dreyer, H. & Wynn, M. (2016), Tacit and Explicit Knowledge in Software Development Projects: A Combined Model for Analysis. *International Journal on Advances in Software*, 9 (3/4), 154-166.

E

- Eagleton, T. (1983). *Literary theory: An introduction*. Oxford: Basil Blackwell.
- Edmondson, A.C., Winslow, A.B., Bohmer, R.M.J., & Pisano, G.P. (2003). Learning how and learning what: effects of tacit and codified knowledge on performance improvement following technology adoption. *Decis. Sci.* 34 (2): 197-223.
- Edvinsson, L. & Sullivan, P. (1996). Developing a model for managing intellectual capital. *European Management Journal*, 14 (4): 356-364.

F

- Faraj, S. & Sproull, L., (2000). Coordinating expertise in software development teams. *Management Science* 46 (12), 1554-1568.
- Füller, K., Ramanath, R., Böhm, M., & Krcmar, H. (2015, March). Decision Support for the Selection of Appropriate Customer Integration Methods. Paper presented at the 12th International Conference on Wirtschaftsinformatik, Osnabrück, Germany.

G

- Gourlay, S. (2002, April). Tacit knowledge, tacit knowledge or behaving? Paper Presented at the Third European Conference on Organizational Knowledge, Learning, And Capabilities. Athens.
- Granovetter, M.S. (1973). The strength of weak ties. *American Journal of Sociology*, 78 (6): 1360-1380.
- Greene, M. (1997). The lived world, literature and education. In D. Vandenberg (ed.), *Phenomenology & education discourse* (pp. 169-190). Johannesburg: Heinemann.

H

- Hammersley, M. (2000). *Taking sides in social research*. London: Routledge.
- Handy, C. (1994). *The Age of Paradox*. Boston: Harvard Business School Press.
- Heemstra, F. J., & Kusters, R. J. (1996). Dealing with risk: A practical approach. *Journal of Information Technology*, 11, 333-346.

Holloway, I. (1997). *Basic concepts for qualitative research*. Oxford: Blackwell.

Huawei, L., Xiaoyong, D., & Meiyun, Z. (2002). *Knowledge management in theory and practice*. Beijing: Huayi Press.

Hycner, R. H. (1999). Some guidelines for the phenomenological analysis of interview data. In Bryman, A., Burgess, R. G. (Eds.), *Qualitative research Vol. 3*, (pp. 143–164). London: Sage.

J

Jackson, J. (1990). I am a Fieldnote: Fieldnotes as a Symbol of Professional Identity. In *Fieldnotes: The Makings of Anthropology*. Roger Sanjek, ed. (pp. 3-33). Ithaca: Cornell University Press.

Johannessen, J.A., Olaisen, J & Olsen, B. (2001). Mismanagement of tacit knowledge: the importance of tacit knowledge, the danger of information technology, and what to do about it. *International Journal of Information Management*, 21: 3-20.

Johnson, C.; Dowd, T.; Ridgeway, C. (2006). Legitimacy as a Social Process. *Annual Review of Sociology*, pp. 53-78.

Jorgensen, D. (1989). *Participant Observation*. Thousand Oaks, CA: Sage Publications.

K

- Kabat-Zinn, J.; Lipworth, L.; Burncy, R.; & Sellers, W., (1986). Four-Year Follow-Up of a Meditation-Based Program for the Self-Regulation of Chronic Pain: Treatment Outcomes and Compliance. *The Clinical Journal of Pain*; 2: pp. 159–73.
- Kelly, G. A. (1955/1991). *The Psychology of Personal Constructs* (Vols. 1 & 2), Routledge.
- Khosrowpour, M. (2004, May). *Innovations Through Information Technology*. Information Resources Management Association International Conference New Orleans, Louisiana.
- Klein, G. A. (1999). Applied decision making. In P. A. Hancock (Ed.). *Human performance and ergonomics* (pp. 87-107). San Diego, CA: Academic Press.
- Kolb, D. A. (1984). *Experiential learning: Experience as the source of learning and development*. New Jersey: Prentice-Hal.
- Koskinen, K.U. (2000). Tacit knowledge as a promoter of project success. *European Journal of Purchasing & Supply Management*, 6: 41-47.
- Koskinen, K.U., Pihlanto, K. & Vanharanta, H. (2003). Tacit knowledge acquisition and sharing in a project work context. *International Journal of Project Management*, 21: 281-290.
- Koskinen, K.U., & Vanharanta, H. (2000). Tacit knowledge as part of engineers' competence. In: *Proceedings of Extra Skills for Young Engineers*. Maribor, Slovenia.
- Koskinen, K.U., & Vanharanta, H. (2002). The role of tacit knowledge in innovation processes of small technology companies. *International Journal of Production Economics*, 80: 57-64.
- Krippendorff, K. (1980). *Content analysis. An Introduction to its Methodology*. Beverly Hills: Sage.

Kruger, D. (1988). An introduction to phenomenological psychology (2nd ed.). Cape Town, South Africa: Juta.

Kvale, S. (1996). Interviews: An introduction to qualitative research interviewing. Thousand Oaks, CA: Sage.

L

Leonard, D., Barton, G. and Barton, M. (2013). Make Yourself an Expert. Harvard Business Review. Retrieved from:
<https://hbr.org/2013/04/makeyourself-an-expert>

Leonard, D & Sensiper, S. (1998). The Role of Tacit Knowledge in Group Innovation, California Management Review, Vol. 40 (3): 112-132.

Leonard, D. and Swap, W. (2014). Critical Knowledge Transfer. Harvard Business Review Press.

Lewis, K. (2003). Measuring transactive memory systems in the field: scale development and validation. Journal of Applied Social Psychology, 33(4): 587-604.

Lewis, K. & Herndon, B. (2011). Transactive memory systems: current issues and future research directions. Organizational Science, 22(5): 1254-1265.

Liang, D.W., Moreland, R., & Argote, L. (1995). Group versus individual training and group performance: the mediating role of transactive memory. Pers. Soc. Psychol. B., 21(4): 384-393.

Lyles, M.A., & Schwenk, C.R. (1992). Top management, strategy, and organizational Knowledge structures. Journal of Management Studies, 29(2): 155-174.

Lynn, L.E. (1998). The Management of Intellectual Capital: The issues and the practice. Management Accounting Issues Paper 16 Management Accounting Practices Handbook. Society of Management Accountants of Canada, Hamilton, Ontario.

M

MacQueen, K. M., McLellan, E., Kay, K., & Milstein, B. (1998). Codebook development for team-based qualitative research. *Cultural Anthropology Methods Journal*, 10(2): 31-36.

Mason, J. (1996). *Qualitative researching*. London: Sage.

Maypole, J., & Davies, T. G. (2001). Students' perceptions of constructivist learning in a community college American History II. *Community College Review*, 29(2): 54-80.

McAdam, R. Mason, B. & McCrory, J. (2007), Exploring the dichotomies within the tacit knowledge literature: towards a process of tacit knowing in organizations, *Journal of Knowledge Management*, 11(2): 43-59.

McAfee, A. (2003). When too much IT knowledge is a dangerous thing. MIT Sloan Management Review. Retrieved from:
<http://sloanreview.mit.edu/article/when-too-much-it-knowledge-is-a-dangerous-thing/>

Meherabian, E.W. (1971). *Silent Messages*. Belmont, CT: Chandler.

Miles, M. B., & Huberman, A. M., (1994). *Qualitative Data Analysis: An Expanded Sourcebook*. Thousand Oaks, CA: Sage Publications.

Min-Xuan, Z. (2003). Tacit knowledge and the apparent possibility of tacit knowledge. *Global Education*, (8): 15-23.

- Minichiello, V., Aroni, R., Timewell, E., & Alexander, L. (1990). In-depth Interviewing: Researching people. Hong Kong: Longman Cheshire Pty Limited.
- Mohammed, S., & Dumville, B. C., (2001). Team mental models in a team knowledge framework, *Journal of Organizational Behavior*, 22: 89-106.
- Moreland, R.L. (1999). Transactive memory: learning who knows what in work groups and organizations. *Shared cognition in organizations: the management of knowledge*. Eds. J.M. Levine and D.M. Messick (eds.). (p.331). Mahwah, NJ: Erlbaum.
- Moreland, R.L. & Myaskovsky, L. (2000). Exploring the performance benefits of group training: transactive memory or improved communication? *Organ Behav. Hum. Dec.* 82(1): 117-133.
- Mouton, J. & Marais, H.C. (1990). Basic concepts in the methodology of the social sciences (Revised ed.). Pretoria, South Africa: Human Sciences Research Council.

N

- Nader, L. (1986). From Anguish to Exultation. In *Women in the Field: Anthropological Experiences*, 2d ed. Peggy Glode, ed. pp. 97-116. Berkeley: University of California Press.
- Nahapiet, J. & Ghoshal, S. (1998). Social capital, intellectual capital, and the organizational advantage. *The Academy of Management Review*, 23(2): 242-266.
- Nerur, S., & Balijepally, V. (2007). Theoretical reflections on agile management methodologies. *Communication ACM* 50: 79-83.
- Nickerson, R. (1998) Confirmation Bias: A Ubiquitous Phenomenon in Many Guises. *Review of General Psychology*, 2(2): 175-220.

Nonaka, I. & Konno, N. (1998). The concept of 'Ba': building a foundation for knowledge creation. *California Management Review*, 40(3): 40-54.

Nonaka, I. & Takeuchi, H. (1995), *The Knowledge - Creating Organization* Oxford University press.

Nonaka, I. & Teece, D. (2001). *Managing industrial knowledge: creation, transfer and utilisation*. London: SAGE Publications.

O

Orlikowski, W. J., & Baroudi, J. J. (1991). Studying information technology in organizations: Research approaches and assumptions. *Information Systems Research*, 2(1): 1-28.

P

Paul, B. (1953). Interview Techniques and Field Relationships. *Anthropology Today*. A. L. Kroeber, ed. Pp. 430-451. Chicago: University of Chicago Press.

Patton, M.Q. (2002). *Qualitative Research and Evaluation Methods*. Thousand Oaks, CA: Sage.

Polanyi, M. (1958). *Personal knowledge: towards a post-critical philosophy*. Abingdon, UK: Psychology Press.

Polanyi, M. (1966), *The Tacit Dimension* Routledge and Kegan Paul, London, UK.

Project Management Institute (2008). *Standard A Guide to the Project Management Body of Knowledge (PMBOK(R) Guide) – 4th ed.* Newtown Square, PA, United States.

Punch, K.F. (1998). *Introduction to Social Research: Quantitative and Qualitative Approaches*. Thousand Oaks: Sage.

R

Reber, A.S., (1995). *Implicit Learning and Tacit Knowledge: An Essay on the Cognitive Unconscious*. Oxford University Press, New York.

Reber, A.S. (1996). *Implicit learning and tacit knowledge*. Oxford, UK: Oxford University Press.

Reed, S., Hock, H., Lockhead, G., (1983). Tacit knowledge and the effect of pattern recognition on mental scanning. *Memory and Cognition*, 11(2): 137–143.

Robertson, M.H.B., & Boyle, J.S. (1984). Ethnography: Contributions to nursing research. *Journal of Advanced Nursing*, 9, 43-49.

Robinson, D., & Reed, V. (Eds.). (1998). *The A – Z of social research jargon*. Aldershot, UK: Ashgate.

Robson, C. & McCartan, K. (2011). *Real World Research*. 4th Edition. West Sussex, UK: John Wiley & Sons Ltd.

Rogers, E., 1995. *Diffusion of Innovations*. Free Press, New York.

Ryan, GW. & Bernard, HR. (2000). Data management and analysis methods. NK Denzin & YS Lincoln (Eds.) *Handbook of qualitative research* (2nd Edition). pp. 769-802. Sage Publications.

Ryan, S. & O'Connor, R. (2009). Development of a team measure for tacit knowledge in software development teams. *The Journal of Systems and Software*, 82: 229-240.

Ryan, S. & O'Connor, R. (2013). Acquiring and sharing tacit knowledge in software development teams: an empirical study. *Information and Software Technology* 55: 1614-1624.

S

Salancik, G., & Pfeffer, J. (1978). A social information processing approach to job attitudes and task design. *Administrative Science Quarterly*, 23: 224-253.

Sandelowski, M. (1995). Qualitative analysis: What it is and how to begin. *Research in Nursing and Health*, 18: 371-375.

Saunders, M., Lewis, P. & Thornhill, A. (2012). *Research Methods for Business Students* 6th edition, Pearson Education Limited.

Sayed-Ikhsan, O., & Rowland, F. (2004). *Journal of Knowledge Management*, 8: 95-111.

Schmidt, R., Lyytinen, K., Keil, M., & Cule, P. (2001). Identifying software project risks: An international Delphi study. *Journal of Management Information Systems*, 17(4): 5-36.

Seligman B.Z. (1951). *Notes and Queries on Anthropology*. 6th ed. London: Routledge & Kegan Paul.

Sharp, H., & Robinson, H. (2010). Three C's of agile practice. *Agile Software Development: current research and future directions*. T. Dingsoyr et al. (eds.). (pp.61-85). Berlin: Springer.

Siau, K. & Erickson, J. (2008). *Advanced principles for improving database design, systems modelling and software development*. New York: Information Science Reference.

Spradley, J. P. (1979). *The ethnographic inter- view*. New York: Holt, Rinehart & Winston.

Spradley, J.P. (1980). Participant Observation. New York: Holt, Rinehart and Winston.

Sternberg, R. J., Forsythe, G. B., Hedlund, J., Horvath, J. A., Wagner, R. K., Williams, W. M., Snook, S. A., & Grigorenko, E. L. (2000). Practical Intelligence in Everyday Life. Cambridge, UK: Cambridge University Press.

Sternberg, R. J., & Horvath, J. A. (Eds.). (1999). Tacit knowledge in professional practice: Researcher and practitioner perspectives. Mahwah, NJ: Lawrence. Erlbaum.

Stewart, T.A. (2010). Intellectual capital: the new wealth of organization. New York: Random House.

Stone, P. J., Dunphy, D. C., Smith, M. S., & Ogilvie, D. M. (1966). The General Inquirer: A computer approach to content analysis. Cambridge: MIT Press.

Stones, C. R. (1988). Research: Toward a phenomenological praxis. In D. Kruger (Ed.), An introduction to phenomenological psychology, 2nd ed. (pp. 141-156). Cape Town, South Africa: Juta.

Strauss, A. L., & Corbin, J. (1990). Basics of qualitative research: Grounded theory procedures and techniques. Newbury Park, CA: Sage.

T

Toffler, A. (1990). Powershift: Knowledge., Wealth and Violence at the Edge of the 21st Century. New York, United States: Bantam Books.

Tsoukas, H. (2003). Do we really understand tacit knowledge? The Blackwell handbook of organizational learning and knowledge management. M. Easterby-Smith & M. Lyles (eds.). (pp.410-427). Boston, MA: Blackwell.

Tsuchiya, S. (1993). Improving knowledge creation ability through organizational learning. IIIA, Proceedings of International Symposium on the Management of Industrial and Corporate Knowledge, 93: 87-95.

U

Ulrich D (1998). Delivering results, a new mandate for human resources Professional. Harvard Business Review. Jan-Feb; 76(1) : 124-34.

V

Von Krogh, G., & Roos, J. (1996). Five claims of knowing. European Management Journal, 14: 423-426.

Von Krogh, G., Roos, J. & Slocum, K. (1994). An essay on corporate epistemology. Strategic Management Journal 15: 53-71.

W

Waldron, V.R. (1986). Interviewing for Knowledge. IEEE Transactions on Professional Communications, PC 29(2): 31-34.

Walsham, G. (1995). Interpretive case studies in IS research: nature and method. European Journal of Information Systems, 4: 74-81.

Wang, V.C.X. (2009). Handbook of research on e-learning applications for career and technical education. New York: Information Science Reference.

Wegner, D.M. (1987). Transactive memory: a contemporary analysis of the group mind. *Theories of group behavior*. B. Mullen & G.R. Goethals (eds.). (pp.185-208). New York: Springer.

Welman, J. C., & Kruger, S. J. (1999). *Research methodology for the business and administrative sciences*. Johannesburg, South Africa: International Thompson.

Wijetunge, P (2012). Organizational storytelling as a method of tacit knowledge transfer: Case study from a Sri Lankan university, *International Information & Library Review*, 44(4): 212-223.

Winter, S. (2016) Tacit Knowledge in *The Palgrave Encyclopedia of Strategic Management*. (pp. 1-3). New York: Springer.

Y

Yin, R. K. (1994). *Case study research: Design and methods*, 2nd ed. Newbury Park, CA: Sage Publications.

Z

Zhang, Y., & Wildemuth, B. M. (2009). Unstructured interviews. In B. Wildemuth (Ed.), *Applications of Social Research Methods to Questions in Information and Library Science* (pp.222-231). Westport, CT: Libraries Unlimited.

Zhi-Guo, L. & Cui-Jian, J. (2012). Improve technological innovation capability of enterprises through tacit knowledge sharing. *Procedia Engineering*, 29: 2072-2076.

Appendix

The appendix consists of three examples of excel sheets used to document the research cycles. The first excel sheet was categorized in accordance to recording dates of the meetings. Here a general overview of discussed topics and whether tacit knowledge could be found was documented. The second cycle was then categorized into the topics discussed and sub categories of the dates were made. This helped identify similar discussions over time. Finally, the third research cycle was categorized into tacit knowledge categories, taking the dates and topics into account.

Not all the excel pages are in the appendix since the importance of the general categorization and its evaluation in the thesis is sufficient to understand the coding process. In addition, within the three research cycles the data was reprocessed and would simply add a repetitive nature to the material for the reader.

Evaluation 1. Cycle Excel Sheet Example

	A	B	C	D	E	F	G	H	I
1	Date	Duration	General	Subjects	Detail	Comments	General Action	Tact Knowledge	Outcome for Others
2	04/03/2013	00:09	HR A asks for CL A Phone Call	Payroll	Pramilla - Licenses	Payroll Manager Role			
3				Project Management	SD A no update		nervous, not done		
4				Sick Pay Policy	CL A info - different kinds				
5				License Issuing	Emails to CL A - Reissue of license after 6 days	NetSuite, needs proper filing. Problems of starting.			
6									
7				Scheduling between people while waiting for Conference Call					
8									
9	04/03/2013	01:38	Conference Call	License Issuing	CL A received lots of emails	glad there are no more coming in	understanding - CL A; SD A nervous laughter		problem solving and excuse for CT
10				Account Set-Up	Issue of only CL A having access				
11				Payroll	Pramilla Management Accountant, Access?	CL A - no change access; records with pay and benefits. CT not programmed Payroll Manager yet, pre question.	learning for future	EK	SD A, HR A learn
12					SD A does not want to switch off Paper and go live directly with CT.	HR side no problem for switch; paper different system to CT.	selling		
13						CL A asks to leave paper till end of fiscal year (June in UK) year. SD A yes.	uncertainty of software.		CL A learns
14						SD A. Process of going Live (at this point 1.4), leave both software running and compare reports.		EK to TK	CL A learns
15				CL A asks for trial monthly run, end of march, April		SD A passes question on to SD B	no knowledge transfer		
16				Sick Pay	No up to date sick pay policy HR consultancy	6 weeks ago update Customer, when update for new long term sick pay.	receiving document for future	Triggered Response	SD A, HR A learn
17				Addresses and Maps	CL A asks for Map feature for address	SD A - separate feature to map addresses - payable	problem - solution	Explicit Element	CL A learns
18			00:11:30	Call end		SD A relief call ended			
19					Discussion about Payroll till end of April	SD A wants to keep both up as long as possible	discussion		
20					HR A sees financial aspect - fear of CL A rejecting Payroll simultaneously running on Paper and CT				
21				Working Patterns	SD A explains NetSuite working patterns		learning software	EK to TK	Learning HR A and HR B
22				Employee Record	HR manager access	SD A explaining access for different users	HR A learning	EK to TK	
23				360	show process of appraisals	Rebecca shows appraisals	HR A learning	EK to TK	Group learning (SD A HR & HR A, HR B software view)
24				Disciplinary and Grievances (D&G)	Ammendements window for disciplinary action	looking at appeals, group talk (union rep), outcome, investigation	learning group HR, software ammerndments	Group TK, learning	
25				Forecasting	Learning time records and forecasting, absences	SD A quick explain, SD B detail	learning software	EK to TK	HR A and HR B learn
26				Allowances	Pav toys, amounts, authorised, ourchases what can they buy:	First Aid, season tube tickets, travel, who approves it, what can they buy:	learnine software	EK to TK	HR A and HR B learn
						25.03.2013	26.03.2013	28.03.2013	02.04.2013
						11.03.2013	12.03.2013	04.04.2013	08.04.2013
						05.03.2013	07.03.2013	11.04.2013	15.04.2013
						04.03.13			
						Overview			

Evaluation 2. Cycle Excel Sheet Example

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
	Date	Record	Enviroment	Person	Subject	Sub	Teaching	Conversation	Social Interaction	Constructive Learning	group	Individual Knowledge	trigger	Decision
1	03/04/2013	01:38	core	SD A	Disciplinary and Grievances	appeal		They are pretty similar as in the appeal process is similar in each incident, it is linked. An appeal is linked to a diciplinary.		x				
2				SD A				Would a capability ever be raised externally?	x					
3								I suppose it could be a customer complaint or something? But then it would be dealt with by HR, so you could simply add the information. It would then just be a complaint, which is then assessed whether it needs to become a disciplinary action	x				x	
4				HR A			Alligations, Response, Investigations			x				
5				SD A	capability									
6				HR A				What is the difference between, investigation by and chairman?	x		x		x	
7				SD A				If it is a servious thing you have a chairman and an investigator.						
8				SD A			companion. Outcome or penalty, notes and dates.			x				
9				HR B				Should we have the notes confidential?	x					
10				HR A				Appeals can only be seen by the HR manager, so it's fine.	x				x	
11				SD A				This is capability, this can be seen by the line manager.	x					
12				HR A				Oh really! So HR B you're right we do need a confidential tab.	x			x	x	
13				SD A				Let's change it then!	x					x
14				SD A			x	Grievance comes in and doesn't go out, I wonder if it is the right term? Grievance Response Sent Date.	x				x	
15				HR A				Yea, it would be grievance recieved, and then response sent and then the date. An appeal would have the same process, just that there is an appeal date.	x		x			
16				SD A			fills out forms, things that need to be added							
17				SD A	grievance		Similar to the other one.			x			x	
18				SD A				Is there a penalty to a grievance?	x					
19				HR A				No, not really. Unless it was a disciplinary case in account to grievance, then I suppose it would all be wrapped together, but not really no. I would keep that in the disciplinary section.	x					
20				SD A				Would you have the penalty?	x		x		x	
21				HR A				No, the penalty would start the disciplinary process. Wait no! There might be a penalty later during the process, they might merge within.	x					
22				SD A	disciplinary		Walkthrough of a disciplinary action- stages							
23				HR A				Can we change the topics within?	x	x			x	
24				SD A				Yes you can			x			
25				SD A			Walkthrough of a disciplinary action							
26				SD A				How many stages do you have?	x					
27				HR A				3.	x		x			
28				SD A				i think that's standard.	x					
29				SD A				The key thing we want there is on informal you loose termination date and suspension		x	x			
<div><div><div>360 Feedback</div><div>Payroll</div><div>General</div></div><div><div>Time</div><div>Recruitment</div><div>EE Centre</div></div><div><div>Disciplinary and Grievances</div><div>Training</div><div>Additional</div></div></div>														

Evaluation 3. Cycle Excel Sheet Example

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	Date	Record	Environment	Person	Subject	Sub	Knowledge aprial	Conversations	Social Interaction	Constructive Learning	group	internalization	trigger	Decision	Recall	EK -Doc	Visual	Conversation	Recall	CL	Anticipation
2	04/03/2015	00:09	Briefing before	HR A	Access			What access does CL.A want for CL.B?													
3		01:38	Conference Call	HR A				CL.B, who've been come aware of over the past couple of weeks, who I think is relatively new?													
4				CL.A				Yes, she's the replacement of the management accountant in the finance team. (.) She has been delegated payroll responsibility.	x	x			x								
5				HR A				Once Software Development Company is set up, what kind of access do you envision CL.B having?													
6				CL.A				Well, I don't want her to have any change access. It is useful for her to be able to access records related to pay and/or benefits.													
7				HR A				There is a role on Software Development Company which is called payroll manager or payroll administrator, which in our working we haven't got there yet but I thought it would be helpful to know which kind of level of access you would want her to have so I can bear that in mind once we start looking at that access role.	x												
8				CL.A				that sounds good.	x												
9	04/03/2015		Explation	Software (one)	SD A	Additions and Deductions	Pay by period	So they might have a monthly london weighting allowance. What do you pay by period?	x								x	x			
10				HR A				They have clothing allowance. first aid allowance.	x												
11				SD A				So it those sort of things. So it has a name, pay by period name, it has a pay type, it has a period it can fall into. It has to be authorized.	x												
12				HR A				Every period?	x									x			
13				SD A				Every payment has to be authorized. Sorry yes, it is authorized on their account and then its generated into weekly or monthly payroll as it gets signed off.	x	x											
14				HR A				Would you only put in payments for that month or put in something...	x												
15				SD A				...you put it in as a go ahead, so when you set it up you select if it is set up for once or if it runs every month or... For example season tickets run over 10 or 12 months.	x	x											
16	04/03/2015			SD A	Pensions			Multiple Pensions. Order of priority. So when they run out of money, this one comes first, this one comes next...	x												
17						Attachment of earnings		Say you are on 500 Pounds a week and you get an attachment of earnings because you failed to pay your child support. So the attachment will have top priority. There is a level in which deductions should stop.	x	x											
18				Rachel				Sorry can you just go back to the pensions type.	x								x				
19				SD A				yes.	x												
20				Rachel				Just wanted to see where I can attach the file.	x												
21	04/03/2015			SD A				I think this needs a real thorough look, I am just skimming through it.	x												
22				SD A	Salary Rates			How much is this person going to get for their basic rate. Time rate 1, 2, ...	x	x											
23				SD A				Particular where people get paid by the hours, overtime hours that sort of thing.	x	x											
24				SD A				Salary rate tie back.	x	x											
25				HR A				I clarified with down and they don't do tie backs.	x												
26	04/03/2015	00:07		SD A	Payments			Is it a standard wage? You can have multiple standard wages such as london living wage.	x												
27								You can put pay on hold. So you know when Customer... just going to get Hanna up to speed.	x	x							x				
28								Explanation of how Customer gets architects together for a particular project.	x	x											
29								You can put a start and end date on hold. Pay date might not go through holidays.	x	x							x				
30				HR A				So that is going to be the annual basic pay, sorry, the FTA isn't it? Oh no its going to be FTM.	x												
31				SD A				yes.	x												
32				HR A				because over here you have the percentage hasn't you. So will it work out?	x												
33				SD A				I don't know, we need to ask ludith.	x												
34				HR A				Because otherwise there is a lot of room for error.	x												
35				SD A				The pro rata bit didn't work, the res did.	x												
36				SD A				The standard hours need to be calculated to see hourly rate by default. (on screen)	x												
37	05/03/2013		Conference	HR A	Payroll			We would need to understand from your perspective (julia) what we need with regards to the payroll report?	x	x								x			
38								Currently, when you receive the payroll data from Customer... you get the paper spread sheet. Does that get important or manually keyed in?	x												