Tacit and Explicit Knowledge in Software Development Projects: Towards a Conceptual Framework for Analysis

Hanna Dreyer
The Business School
University of Gloucestershire
Cheltenham, UK
Dreyer.Hanna@gmail.com

Martin Wynn
School of Computing and Technology
University of Gloucestershire
Cheltenham, UK
MWynn@glos.ac.uk

Gerald Robin Bown
The Business School
University of Gloucestershire
Cheltenham, UK
GRBown@glos.ac.uk

***Abstract* – The management and delivery of software development projects remains a key business activity in many industries. Although the advent of packaged software products has reduced the incidence of in-house development, bespoke software is still important for some industrial sectors - notably in the finance, defence and security industries. Despite the recognized criticality of software project success for organizations, a considerable proportion of projects continue to either not meet their due dates, exceed budget, do not deliver to specification, miss quality targets, or do not meet customer requirements. Software project failure – be it bespoke products or the implementation of commercially available packages - remains an area of considerable interest in contemporary software project management literature, and the management and transfer of knowledge within both these types of project is a key dimension and driver of project outcomes. This paper examines how knowledge definition and management can be applied within a conceptual framework to improve software development project outcomes.**

Keywords – software development; tacit knowledge; explicit knowledge; project management; conceptual model

I. INTRODUCTION

This research aims to uncover the connection between tacit and explicit knowledge in software development projects. McAfee [1] pointed out the dangers of differing interpretations of checklists in attempting to achieve success in software projects. He concluded that "the successful leadership of an IT implementation will continue to be a subtle craft", in which the appropriate interpretation and application of knowledge are key. Tacit knowledge is difficult to articulate but is, according to Polanyi [2] [3], the root of all knowledge, which is then transformed into explicit, articulated knowledge. The process of tacit to explicit knowledge transformation is therefore a key component of software development projects.

This introductory section is followed by four further sections. Section 2 contains a discussion of the theoretical framework for this paper. Then, in Section 3, the research methodology is discussed. Section 4 outlines progress to date on developing a conceptual framework for analysis; and finally, Section 5 looks at how this research can be further progressed.

II. THEORETICAL FRAMEWORK

Knowledge is conceived of as a collection of information or usable data that is put into context, in order to comprehend how something works. Unlike most goods, knowledge does not have a physical form and cannot therefore be touched. The question remains as to what extent it can be spoken. The relationship of the whole, the entire focal point, to the parts, the detail, is a partnership where one part has larger intensity than the other [3]. Developing Polanyi's work, Nonaka and Takeuchi [4] suggest that tacit knowledge can be converted into explicit knowledge, and therefore expressed in words or numbers. Human knowledge transfer is one of the greatest challenges in contemporary society, due to knowledge often being inaccessible. Its initial creation, transfer, utilization, and storage are often problematic, both conceptually and operationally.

Nonaka and his co-authors' studies on knowledge, although influential, still leave a number of unanswered issues, such as those concerned with group tacit knowledge and explicit knowledge [5]. There have been many critics of tacit knowledge since Polanyi's Personal Knowledge was published in 1962 [6]. Tacit knowledge is personal knowledge that an actor knows he has, but which he cannot describe in terms other than its own performance. An opinion is that "there can be no doubt that Personal Knowledge comes at us with its rhetoric all out of focus. The lack of clarity can be said to be due to its heterology; "it is a mixed bag" [7]. Critically challenging linguistic based knowledge with the idea that "we can know more than we can tell" [2], tacit knowledge still needs more exploring in order to be verified. These elements provide the rationale for this research and define the space it seeks to address.

Knowledge creation is a continuous, self-transcending process by which one can transcend the boundary of the old self into a new self by acquiring a new context, a new view of the world and new knowledge [8]. The raw product of knowledge is information, which differs greatly from knowledge due to it not being involved with a specific context. "Without context, it is just information, not knowledge" [8]. Having a lot of information will not serve a company unless it has been evaluated, put into context and agreed that it is a justified true belief.

From individual knowing, the development of shared understanding is important in any joint enterprise. Knowledge is a key asset in companies; it can serve as a

 ISBN: 978-1-61208-386-5

basis to build a greater knowledge base and can therefore provide a competitive advantage [8]. In differentiating between tangible assets, which can only be used by one party, and intangible knowledge assets, which can be shared and used by several parties, the utilization, transfer and storage of knowledge becomes a core issue.

Social network analysis plays a significant role when analyzing the process of knowledge creation within a group. Some authors [9] have focused on ways network analysis can be used to identify key players in diffusion networks. "Diffusion of innovations looks at the process by which a new technology or idea gets adopted by a given community" [9]. Furthermore, Prell [10] assesses ego networks, an individual's network, and the influence it has on their alter-egos. These tools aim to help explore the interactions of project members and their positioning within the group in order to evaluate knowledge inputs and outputs.

The environment of this research is a software development project, which influences tacit and explicit knowledge through the time restricted character of projects; the clearly defined beginning and ending [6]. As previously stated, human knowledge transfer is one of the greatest challenges in today's society; not having sufficient time to articulate tacit and explicit knowledge affects project success. In addition, work is allocated and distributed throughout the different actors of the project, which results in the scattering of knowledge throughout project members.

III. METHODOLOGY

The research sets out to identify the impact of tacit and explicit knowledge transferred during software development projects. An inductive, exploratory, qualitative methodology is being applied in order to validate the tacit knowledge spectrum in software development projects. The philosophical foundation of this study is based on the ontology of subjectivism, while the epistemological position is interpretivism. The researcher is centrally involved in the phenomena being studied, and is a key player in the process of data collection and analysis to answer the research questions. In terms of the methodological approach, a case study method is adopted (see Figure 1); this is deemed appropriate given the embedded nature of the study. If a case study strategy incorporates multiple cases, then the resulting data can provide greater confidence in the research findings [11]. Data will be generated through unstructured interviews, and will therefore be accessed in a narrative form [8]. Personal reflection on the data collected during the meetings, as well as participant observation, will also be part of the methodology. The study sets out to evaluate three software development projects as case studies in different companies.

Seeing the company as a contextualized culture, and the participant observation as performative auto ethnography [12], will aid the conduct of this inductive research. Knowledge is assessed by textual forms, empathetic epistemology of critical and co-present reflection with others, transformation to the dominating system, and social as well as linguistic effects with others. This will result in a contextualized conceptual model, proposed for the analysis of tacit and explicit knowledge in software projects.

As tacit knowledge does not have a physical form, the information passed on throughout the project needs to be interpreted in a qualitative form. Generating data through unstructured interviews, participant observation and personal experience will help identify the tacit knowledge within the data. Pre-interviews with the key players in the project, as well as short interviews to track any change of view, and a final conversation, will be part of the interview data generation process.

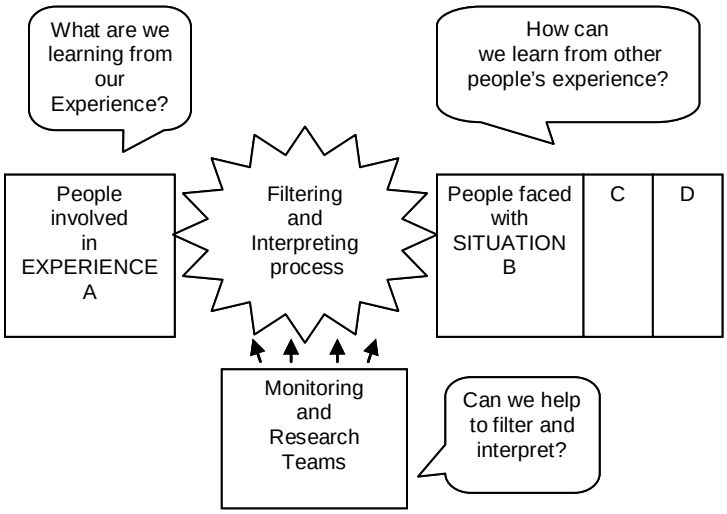

Figure 1. Case Study Research [13]

The aim is to find information in the data that will be collected in over 30 hours of meetings from the first case study. Following this, the results are to be refined through two more software development projects in other companies, which will each consist of 5 hours of meetings. Follow up meetings with several project members will be used to validate the data, and thereby develop the tacit knowledge spectrum in software development projects. This will be done through the analysis and interpretation of the data collected during these meetings. Software development meetings are recorded and transcribed according to the importance of the information; an assessment of the exchanged tacit knowledge in relation to the tacit knowledge spectrum will be produced. The aim of the follow up meetings is to confirm the themes revealed in the data gathered during the meetings. Validation, elaboration, as well as evaluation of the previously found themes should result. Finding key players, and therefore the key knowledge sources, through participant observation will support the evaluated data from the meetings. This should, in the future, aid researchers as well as managers to understand and further develop tacit knowledge in the work place, and in particular in software development projects.

In summary, the main elements of the research method and design are:

1. Qualitative exploratory research
2. Inductive research

 ISBN: 978-1-61208-386-5

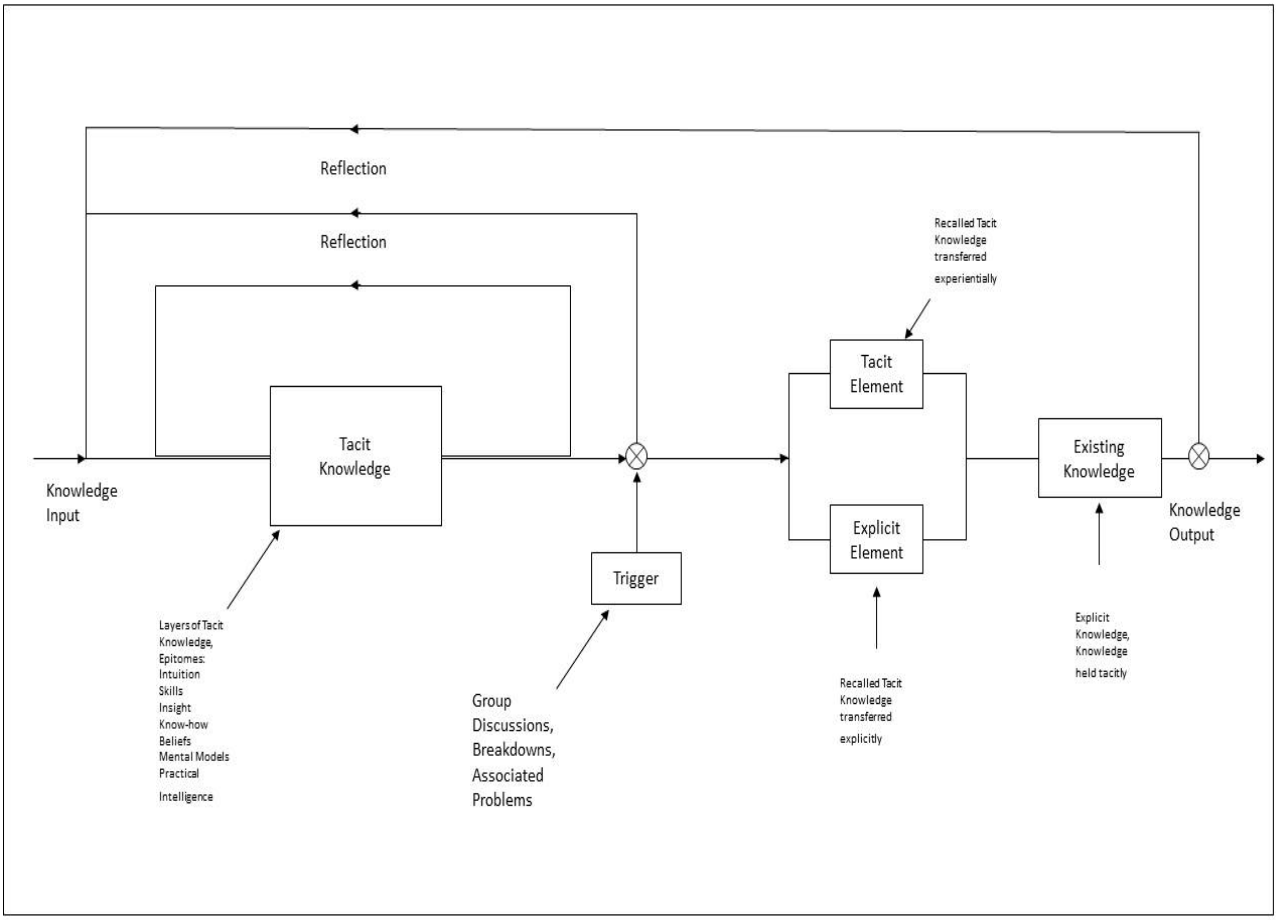

Figure 2. The Tacit Knowledge Specrum [15]

3. Participant observation
4. Personal reflection
5. Unstructured interviews .

Bias will be eliminated through personal reflection and constant validation of the data gathered. Once the emergent themes are found, the interviews will be constructed and later held with the project members. Interviews with the project participants will be held, in order to validate the interpretations and data garnered from the meetings. Through replication logic, external validity will be ensured; internal validity is not needed in exploratory research.

IV. CONCEPTUAL FRAMEWORK DEVELOPMENT

Most of the research in the field of software development projects is quantitative and does not take into account the qualitative inductive aspects of the topic. The main focus of the study is to qualitatively understand the transfer, as well as the impact, of tacit to explicit knowledge and its effect on the development of new software. The aim is to create a model that supports future software development projects in their tacit to explicit knowledge transfers.

A clearly defined beginning and ending, as well as the work to be done, are the main trademarks of projects. The Office of Government Commerce defines a project as “a temporary organization that is needed to produce a unique and pre-defined outcome or result, at a pre-specified time, using predetermined resources” [14]. Therefore, projects are time restricted to a larger extent than day-to-day business. This directly affects software projects since data generation has to be done within the scope of the project and cannot be repeated, due to their unique and time restricted nature. Project failure or delay is often due to poor communication, which is closely related to the exchange of knowledge. Another challenge of projects is the storage and communication of knowledge resources during and after completion. With the advent of email and the internet, knowledge is now mainly transferred through electronic devices in information technology projects, which has greatly changed the information flow. This influences the impact of tacit and explicit knowledge, which, due to it being unarticulated, often remains hidden. In addition, project environments are dismantled, and knowledge resources are often lost once a project is completed. These issues demonstrate the specific character of projects and their long

term benefits, in particular when they concern software development.

The recent work of Clarke [15] puts forward a model of a tacit knowledge spectrum developed from a study of three companies (see Figure 2). Using Clarke's model to test tacit and explicit knowledge provides a provisional conceptual framework to help the research analysis process. Approaching theory as "a way of seeing and thinking about the world rather than an abstract representation of it" [16] will set the data into perspective. Using Clarke's model, and the underlying theory of tacit knowledge, creates the opportunity to investigate theory in the work place. This is based on the three functions of directing attention, organizing experience and enabling useful responses [16] [17].

Clarke's model will help to sensitise and explore the use of tacit and explicit knowledge in software development projects. The research aims to contribute knowledge in the area of knowledge management by increasing awareness about the tacit and explicit knowledge transferred, and its impact on such projects. Given the time and budget constraints normally imposed on software development projects, the goal is to aid project managers and team members in their future planning, as well as improve the transfer of knowledge and information during a project. Through an analysis of existing literature, allied to empirical data and observations in large project environments, this research will look to develop Clarke's conceptual framework and answer the following research questions:

- What is the current understanding of knowledge exchange in software development projects?
- How can tacit and explicit knowledge be recognised and evaluated in software development projects?
- To what extent does non-communicated tacit and explicit knowledge amongst team members influence the project and its acceptance?
- Can tacit and explicit knowledge be better harnessed through the development of a conceptual model for use in software development projects?

This approach assumes that it is feasible and sensible to cumulate findings and generalize results to create new knowledge. The concepts of tacit and explicit knowledge will be analysed in primary research case studies. The key assumption that there is a "trigger", that acts as a catalyst for the recall and transfer of different knowledge elements, will be examined in software development projects. The basic conceptual framework will build on Clarke's Tacit Knowledge Spectrum and will be developed further in the light of further literature analysis and first-hand project research.

V. CONCLUDING REMARKS

Peter Drucker used to tell his students that when intelligent, moral, and rational people make decisions that appear inexplicable, it's because they see a reality different to the one seen by others [18]. This observation by one of the leading lights of modern management science underscores the importance of knowledge perception and knowledge transfer. With regard to software projects, McAfee [1] noted that "the coordination, managerial oversight and marshaling of resources needed to implement these systems make a change effort like no other". Yet, although software project successes and failures have been analysed within a range of analytical frameworks, few studies have focused on knowledge transfer. This phenomenon requires further research into the interaction and communication of knowledge within and between project teams and their contexts, and this is the purpose of this research, in the specific context of software development. If knowledge can be more successfully harnessed to improve the software development process, it has the potential to significantly enhance eventual project outcomes.

REFERENCES

[1] A. McAfee, "When too much IT knowledge is a dangerous thing," MIT Sloane Management Review, Winter, 2003 pp. 83-89.

[2] M. Polanyi. The tacit dimension. The University of Chicago Press, 1966.

[3] M. Polanyi, Personal Knowledge: Towards a Post-Critical Philosophy. Routhledge , 1998.

[4] I. Nonaka and H. Takeuchi, The knowledge-creating company: How Japanese companies create the dynamics of innovation. Oxford: Oxford University Press, 1995.

[5] Z. Erden, G. von Krogh, and I. Nonaka, The quality of group tacit knowledge, 2008

[6] M. Polanyi,. Personal Knowledge, Psychology Press, 1962.

[7] T. Langford and W. Poteat, " Upon first sitting down to read Personal Knowledge: an introduction". In Intellect and hope: Essays in the thought of Michael Polanyi , 1968, pp. 3-18.

[8] I. Nonaka and D.Teece, Managing Industrial Knowledge, 2001, London: SAGE.

[9] T. W. Valente and R. Davies, "Accelerating the diffusion of innovations using opinion leaders", The Annals of the American Academy of Political and Social Science, vol. 566, 1999, pp. 55-67.

[10] C. Prell, Social Network Analysis: History, Theory and Methodology, 2012, London: SAGE

[11] R.K.Yin, Case Study Research (4th Edn), Sage, 2009

[12] D. S. Madison, Critical Ethnography: Method, Ethics and Perforamce, Second Edition 2012, London: SAGE..

[13] M. Wynn, J.L. Taylor, and J. Overall, "Case study monitoring: an approach to urban management training", Planning and Administration, vol. 9, no. 1, Spring 1982, pp 16-24.

[14] Office of Government Commerce (OGC), Managing Successful Projects With PRINCE2, 2009, London: The Stationery Office/Tso.

[15] T. Clarke, The development of a tacit knowledge specturm based on the interrelationships between tacit and explicit knoweldge. 2010, Cardiff: UWIC.

[16] A. Deetz, Doing Critical Management Research, 2000, London: Sage.

[17] M. Alvesson and K. Skoeldberg, Reflexive Methodology, 2004, United Kingdom: SAGE.

[18] B. Baker, , "The fall of the firefly: An assessment of a failed project strategy," Project Management Journal, vol. 33, no. 3, 2002, pp. 53-57.

 ISBN: 978-1-61208-386-5