Open access • Journal Article • DOI:10.1016/J.JSS.2008.05.037

Development of a team measure for tacit knowledge in software development teams

— Source link

Sharon Ryan, Rory V. O'Connor

Institutions: Dublin City University

Published on: 01 Feb 2009 - Journal of Systems and Software (Elsevier)

Topics: Team software process, Personal software process, Tacit knowledge, Team management and Software development

Development of a Team Measure for Tacit Knowledge in Software Development Teams

Sharon Ryan[1], Rory V O'Connor[2]

[1] University of Greenwich, London, UK, S.Ryan@gre.ac.uk
[2] School of Computing, Dublin City University, Dublin, Ireland, roconnor@computing.dcu.ie

Abstract
In this paper we operationally define and measure tacit knowledge at the team level in the software development domain. Through a series of three empirical studies we developed and validated the Team Tacit Knowledge Measure (TTKM) for software developers. In the first study, initial scale items were developed using the repertory grid technique and content analysis. In Study 2, supplied repertory grids were administered to novices and experts to establish differential items, and Study 3 validated the TTKM on a sample of 48 industrial software development teams. In developing the TTKM we explored the relationships between tacit knowledge, explicit job knowledge and social interaction and their effect on team performance as measured by efficiency and effectiveness. In addition we assess the implications for managing software development teams and increasing team performance through social interaction.

Keywords: Knowledge, Tacit Knowledge, Team Tacit Knowledge, Repertory Grid, Team Performance, Team Management, Agile methods

1 Introduction

The investigation and measurement of tacit knowledge has gained popularity in recent years, particularly in the area of organisational behaviour where it is seen as of increasing importance to economic and organisational competitiveness (Fernie et al., 2003). The tacit knowledge concept was introduced by Polanyi (1966) who described it as knowledge that cannot be articulated, and despite its current popularity tacit knowledge remains difficult to conceptualise and measure. Tacit knowledge has been found to explain individual differences in management effectiveness (Wagner and Sternberg, 1991) in leadership effectiveness (Hedlund et al., 2003) and in team performance (Edmondson et al., 2003). Software development is the focus of this study since it is a knowledge-driven industry which relies on employees' expert knowledge to create a finished product, where this expert knowledge is mostly tacit.

Members of software development teams are considered to be 'intellect workers' (DeMarco and Lister, 1987) or 'knowledge workers' (Drucker, 1993) who are characterised as individuals with high levels of education and specialist skill, combined with the ability to apply these skills to identify and solve problems. Since they work with intangible cognitive processes rather than physical tangibles, the rules for developing tangible goods do not apply. Knowledge, especially tacit knowledge is held in individual minds and is the means of production in software development. The process of developing software involves the tacit coordination of expertise of these team members (Faraj & Sproull, 2000). Knowledge sharing is therefore a key process in developing software, and since expert knowledge is tacit, the acquisition and transmission of tacit knowledge is significant in the development process. Furthermore, development methodologies such as Agile methods rely on the tacit knowledge embodied in the team, rather than writing the knowledge down in plans.

The failure of many large software projects has highlighted the challenges in managing team-based work (Faraj and Sproull, 2000). According to the Standish Group CHAOS report, the majority (53%) of software projects do not meet budget and schedule, function unsatisfactorily, and around 18% are challenged (Standish Group International, 2004). A major study into IT projects in the UK reported that IT project success in the UK was 'between only 16% and 34%' (Royal Academy of Engineering, 2004). Research has shown that human factors rather than technological developments may be the primary issues influencing performance on successful projects (Faraj and Sproull, 2000). Team performance on software development projects is dependent on many different and interacting factors like effective plans, good communication, clear goals etc. In addition, internal group processes (team work) particularly those focussing on the team's relationships, are more likely than technical factors (task work) to be associated with team performance on successful projects (Guinan et al., 1998). Team tacit knowledge is human factor and is thought to be related to team work process of social interaction which is necessary for building relationships.

There exists no field measure which addresses the tacit knowledge construct at the collective team-level. In this paper we operationally define and measure tacit knowledge at the team-level through the development and validation of a domain specific Team Tacit Knowledge Measure (TTKM) for the software development sector. We developed the team tacit knowledge measure through a series of three empirical studies. In developing the TTKM we explored the relationships between tacit knowledge, explicit job knowledge and social interaction and their effect on team performance as measured by efficiency and effectiveness. This work is part of a wider project whose purpose is to model and test the acquisition and sharing of tacit knowledge in software development teams. However, this paper addresses the following questions:

1. *Does team tacit knowledge predict performance in software development teams?*
2. *Is there a relationship between team tacit knowledge and explicit job knowledge?*
3. *Do teams with higher levels of social interaction also have higher levels of tacit knowledge?*

This paper initially describes the issues in defining tacit knowledge at the individual and group levels, before forwarding an operational definition on which the TTKM is based, before outlining the ambiguities associated with the tacit knowledge construct. Social interaction in software teams and its relationship to tacit knowledge is then described with a particular focus on the principles of Agile methods. Approaches to measuring tacit knowledge and the chosen methodology are explained, followed by an in-depth procedure for the three empirical studies which has led to the development of the TTKM. Finally, the results are discussed and conclusions drawn. It is argued that, the outcomes of this research will provide practical applications and benefits to people who work in and manage software development teams and support the principles underlying the Agile approach to software development.

2. Defining tacit knowledge

There has been much debate in the literature as to how tacit knowledge can be operationally defined. Sternberg and his colleagues at Yale University, using what Busch et al. (2003) call the 'Yale group approach' have investigated this concept from a psychological perspective and view tacit knowledge as an aspect of practical intelligence. Sternberg et al. (2000) argue that there are three characteristics of tacit knowledge:

- it is acquired with minimum environmental support;
- is procedural taking the form of 'knowing how' rather than 'knowing that'; and
- is practically useful for the attainment of personal goals.

The development of a team-level definition for tacit knowledge should reflect the social cognitive process which gives rise to shared mental models using groups or teams of people as the unit of analysis, where individual scores on cognitive measures are aggregated. Cannon-Bowers et al. (1993) argue that there are two types of mental models which can shared among team members: The task model which contains information about task procedures, strategies, and other likely scenarios and the team model which refers to knowledge about team-mates' knowledge, skills, abilities, preferences and tendencies

Mindful of the ambiguities associated with defining tacit knowledge, it is necessary to provide some sort of operational definition in order to develop a team measure. This definition would ideally incorporate the individual and social aspects of tacit knowledge thus accounting for the epistemological perspective that knowledge is both personally and socially constructed (Kelly, 1955/1991).

The definition offered by Sternberg et al (2000), is the basic definition used in this study. However, there is one modification, this definition of tacit knowledge is also conceptualised at the group or team level. This is defined as the *aggregation* of individual tacit knowledge to the team level, where different members of the team will possess different aspects of the tacit knowledge. In this paper, team tacit knowledge is defined as: *The aggregation of articulable tacit, individual, goal driven expert knowledge to the team-level where different members of the team possess different aspects of tacit knowledge*. Team tacit knowledge is therefore related to task models and team models and is thought to be acquired through social interaction.

2.1 Ambiguities in conceptualising tacit knowledge

Difficulties in measuring tacit knowledge are directly related to the lack of clarity in conceptualising the construct. In a review of the literature, Gourlay (2006) identified six ambiguities associated with the conceptualisation of tacit knowledge where tacit knowledge:

- is both individual and collective;
- is acquired through experience but also innate;
- is acquired with or without of others present;
- is a form of practical intelligence whilst also being defensive, naïve or belying incorrect theory;
- facilitates routine behaviours whilst also being a source of innovation; and
- may or may not be converted to explicit knowledge.

To develop a measure for tacit knowledge at the team-level, two questions needed to be addressed: *Is tacit knowledge an individual trait or a trait that can be shared by both individuals and groups*? and *Can tacit knowledge be made explicit*?

2.2 Individual versus group tacit knowledge

Much of the literature treats tacit knowledge as an individual level phenomenon where the concept of tacit knowledge is closely related to skill learning (Polanyi, 1966) and to expertise where '*tacit knowledge distinguishes more successful individuals from less practically successful*' (Sternberg et al., 2000, p.105). Von Krogh and Roos (1995) argue that tacit knowledge is an individual characteristic, which is embedded in action in specific contexts. However, according to Grant (1996) there is a capacity for aggregation of tacit knowledge, which reflects the ability of individuals and teams to absorb new knowledge and add it to existing knowledge.

The cognitive properties of groups can differ from those of their participating members (Hutchins, 1991) and Team Mental Model (TMM) theory provides a good theoretical basis for the conception of tacit knowledge at the group level where a team mental model is '*an organized understanding of relevant knowledge that is shared by team members*' (Mohammed and Dumville, 2001, p.89).

2.3 Knowledge conversion through social interaction

The second question of whether tacit knowledge can be made explicit is closely related to the first, since one of the goals of converting tacit to explicit knowledge is to allow the knowledge to be shared in the organisation. According to Nonaka and Takeuchi (1995) who have popularised the term tacit knowledge in their theory of knowledge conversion which posits that tacit knowledge becomes externalised through iterative, face-to-face interaction. In the software domain Busch et al. (2003) used social network analysis (SNA) to examine formal and informal interactions in an IT department (sample size N = 12) and concluded overwhelmingly that tacit knowledge is diffused in human to human interaction. Tacit knowledge may be shared in a number of ways, including mentoring and apprenticeships, but usually involves social interaction.

However, if tacit knowledge cannot be articulated then is it not paradoxical to claim that it can be externalised? For through this process of knowledge conversion we are changing the nature of the construct. It is argued that some tacit knowledge can be articulated (Sternberg et al. 2000; Nonaka and Takeuchi, 1995) and Busch et al. (2003) label this knowledge as articulable tacit knowledge (aTK). Yet others dispute that tacit knowledge can ever be articulated (Polanyi, 1966) and contend that there exists a middle ground between tacit and explicit knowledge called implicit knowledge, which is knowledge that can be articulated, but has not yet been articulated (Baumard, 1999).

Much confusion in the definition of tacit knowledge has ensued from the use of the terms tacit and implicit interchangeably. According to Jha (2002) a clear definition of tacit knowledge showing causal connections and enabling prediction, is a criterion for a rule-following mechanistic conception of scientific investigation, not a philosophical endeavour as forwarded by Polanyi. We argue here that some tacit knowledge in the realm of scientific inquiry can be articulated and therefore transformed into explicit knowledge which may be useful to organisations and for team performance.

3. Social interaction in software development teams

Software development teams have a relatively unique structure, wherein the division of labour among members is highly interdependent. This is mainly to do with the way in which the finished product is produced; the nature of the software development process is such that the product cannot be seen in its progressive development, unlike say building a bridge and this has implications for team members. Developing software involves complex problem solving and decision-making based on previous experience. According to Blackler (1995), these knowledge workers have specific individual expertise embodied into the practical activity-based competencies and skills of key member (i.e. practical knowledge or know how) and embrained knowledge in the conceptual understandings and cognitive skills of key members. The more each individual's knowledge is shared among members of a team, the larger and more dispersed the knowledge base becomes. At the same time team members become more aware of where the expertise is located within the team. In sharing and coordinating expert knowledge each team member will construct their own knowledge personally and socially, through their interactions with one another. In addition the team will develop a TMM about the task and/or the team.

Members of software development teams may informally exchange knowledge, for example; during lunch or over 'water cooler conversations' (Frangos, 2004); and discover new insights as a result of informal social interaction. The social approach is also important because team members individually have a limited capacity for processing information so that, when dealing with complex problems like software development, they can rarely process all the information that would be relevant. Their mental models help team members to select information and to decide what actions are appropriate (Weick, 1979).

3.1 Implications for software development methods

The traditional software development world, characterised by the engineering and process improvement advocates use plan-driven methods which rely heavily on explicit documented knowledge. Plan-driven methods use project planning documentation to provide broad-spectrum communications and rely on documented process plans and product plans to coordinate everyone (Boehm and Turner, 2004). With plan-driven methods, communication tends to be one-way, from one entity to another rather than between two entities. Process descriptions, progress reports, and the like are nearly always communicated as unidirectional flow.

The late 1990s saw something of a backlash against what was seen as the over-rigidity contained within plan-driven models and culminated in the arrival of agile methodologies, who rely heavily on communication through tacit, interpersonal knowledge for their success (Boehm and Turner, 2004). They cultivate the development and use of tacit knowledge, depending on the understanding and experience of the people doing the work and their willingness to share it. Knowledge is specifically gathered through team planning and project reviews (an activity 'agilists' refer to as *retrospection*) and is shared across the organization as experienced people work on more tasks with different people. Agile methods generally exhibit more frequent, person-to-person communication. As stated in the Agile Manifesto, emphasis is given to *individuals and interactions*. Few of the agile communications channels are one-way, showing a preference for collaboration, with stand-up meetings, pair programming, and the planning game being examples of the agile communication style and its investment in developing shared tacit knowledge.

The distinction between 'agile-tacit' and 'plan-driven-explicit' is not absolute (Boehm and Turner, 2004) as the boundaries between traditional methods and agile methods have yet to be firmly established. According to Cockburn and Highsmith (2001) '*what is new about agile methods is not the practices they use, but their recognition of people as the primary drivers of project success, coupled with an intense focus on effectiveness and maneuverability* [sic]. *This yields a new combination of values and principles that define an agile world view*' (p. 122).

According to Chau and Maurer (2004) it is unlikely that all members of a development team possess all the knowledge required for the activities of software development. Therefore different people will possess different aspects of knowledge, as posited by TMM theory which underlines the need for knowledge sharing in software organisations. Chau and Maurer (2004) cite the example of pair programming as used in XP. Pair programming, is a social process and involves two developers working in front of a single computer designing, coding, and testing the software together. During a pair programming session, some explicit but mostly tacit knowledge is shared between the pair. The knowledge shared includes task-related knowledge, contextual knowledge, and social resources. Chau and Maurer (2004) conclude that '*for this reason, the social nature of pair programming made it a great facilitator for eliciting and sharing tacit knowledge*'(p.4).

4. Approaches to Measuring Tacit Knowledge

Individual tacit knowledge has been measured at the articulated level of abstraction using a form of self-report situational judgment tests (Sternberg et al., 2000), experiments in artificial grammar (AG) learning (Reber, 1995) and mental scanning (Reed et al., 1983). Qualitative case studies have also been applied in tacit knowledge sharing (e.g., Desouza, 2003). Team-level tacit knowledge has been assessed by proxy (Edmondson et al., 2003) and using SNA (Busch et al., 2003). In general expert knowledge forms the basis for tacit knowledge measures.

The Yale group approach is based on the principles that novices and experts differ in the amount and organisation of domain specific knowledge and that tacit knowledge can be articulated. Therefore, the more expert like knowledge a person possess, the more tacit knowledge that individual has. This approach uses situational judgement tests which are low fidelity simulations (i.e. stimuli do not closely represent the actual situation). The process of developing a tacit knowledge inventory in this way begins by eliciting experienced-based tacit knowledge from successful practitioners in a particular domain and finishing with a validated and revised instrument. The potential items are selected to yield a measure of underlying domain relevant tacit knowledge (Sternberg et al., 2000). The Yale group has measured tacit knowledge in sales teams, (Sternberg and Wagner, 1988) academic psychology, (Wagner and Sternberg, 1985) managers, (Wagner and Sternberg, 1991) and military leaders (Hedlund et al., 2003). For example in the study by Hedlund et al. (2003) the Tacit Knowledge for Military Leaders (TKML) inventory, consisting of a series of leadership scenarios, was developed to assess the amount of knowledge leaders possess. Three versions of the TKML were administered to a total of 562 leaders at the platoon, company, and battalion levels. At all three levels, TKML scores correlated with ratings of leadership effectiveness from either peers or superiors, and the scores explained variance in leadership effectiveness beyond a test of general verbal ability and a test of tacit knowledge for managers. These results indicate that domain-specific tacit knowledge can explain individual differences in leadership effectiveness and suggest that leadership development initiatives should include efforts to facilitate the acquisition of tacit knowledge.

Critics argue that Sternberg and the Yale group's tests of tacit knowledge do not reveal the strong empirical support they assert (Gottfredson, 2003). Others have concluded that the test is reliable but not a valid measure of success (Taub et al., 2001).

Tacit knowledge and performance have also been measured at the team-level using proxy measures. Proxy measures attempt to address the problem that tacit knowledge is unobservable and hold that tacit knowledge cannot be articulated, therefore, tacit knowledge needs to be measured by substitution. For example Berman et al. (2002) used data from the National Basketball Association (NBA) and argue that their measure is a reasonable proxy for the sort of tacit knowledge at team-level. Years of player team experience was weighted by the minutes played in the games that season by that player and an average was then calculated for each team year. The study found that team success increased as the team's tacit knowledge increased and concluded that tacit knowledge is gained through experience rather than formal study methods and can be acquired at an individual or group level. In another study, Edmondson et al. (2003) used a performance measure of efficiency as the proxy measure for tacit knowledge in cardiac surgical teams in 15 hospitals. Proxy measures are encumbered by issues of construct validity.

Accounting for the criticism levelled against the Yale group and the inherent problems in using proxy measures for team measures, we now outline our approach to developing the TTKM.

5. The Study Method

The rationale and procedure for the development and validation for the TTKM was a knowledge-based approach, whereby experts differ from novices in task performance relative to their domain of expertise. Developing a team-level measure of tacit knowledge for software development teams needs to address a number of issues:

1. Team tacit knowledge is a reflection of domain specific expert knowledge which differentiates novices from experts and reflects practical experience.
2. It should measure the tacit knowledge possessed by all team members, albeit to different degrees in order to aggregate to team-level.
3. The measure can only deal with tacit knowledge at the articulated level of abstraction.

5.1 Synopsis of Research Strategy

The rationale for the development and validation of the TTKM was predominately based on the Yale group's general framework for differentiating between novices and experts but applied the Repertory Grid technique rather than situational judgement tests to measure tacit knowledge of experts. The repertory grid was founded on Kelly's (1955/1991) Personal Construct Theory and is a simple technique for accessing tacit knowledge (Stewart and Stewart, 1981) which helps illuminate personal knowledge and gain access to private worlds (Kelly, 1955/1991). The repertory grid, therefore accesses tacit knowledge at the articulated level (Moynihan, 2002). There are three important constituents to the repertory grid: *elements*, *constructs* and *links*. The repertory grid provides a two-way classification of information in which relationships are uncovered between a person's observations of the world (called elements) and how they construct or classify those observations. These constructs are made up of similarity-difference dimensions or bipolar constructs, describing how some elements are similar and yet different from another. An example of one of the bipolar constructs is 'Innovative project <----> Mundane/Everyday type project'. The third component of the grid links the elements and constructs, where each element is rated on each construct.

5.2 Statistical Justification for Aggregation

At this point, an important question to address is: *Is it appropriate to aggregate (sum) or average team member responses to a meaningful team-level indicator for tacit knowledge*? According to Lewis (2003), items which focus members' responses on the team remain meaningful when aggregated and the aggregate represents the degree to which teams share a situation specific tacit knowledge team model or tacit knowledge task model.

Before aggregating individual responses to the team-level, it was necessary to statistically test the conformity of the level of measurement to the level of the theoretical analysis (Klein et al., 1994). The Inter-rater Agreement (IRA) r_{wg} was used to assess within -team agreement for each team separately (James et al.,1984). A strength of this method is that it compares the responses within a team without including any information from the other teams. Values lie between 0 and 1, with values above .70 being the accepted cut-off point for agreement.

5.3 Overview of Validation Analyses

Tacit knowledge inventories and other situational judgment tests differ from conventional knowledge tests in that items may be poorly defined and are multidimensional in nature drawing on skills, knowledge and abilities (Hedlund et al., 2003). Across an inventory there are diverse areas of knowledge some acquired by the individual, some not, therefore the complexities of the tacit knowledge measures reduces the likelihood of obtaining the same levels of internal consistency as for other traditional knowledge and ability tests. According to Legree (1995) we can expect to obtain alpha coefficients between .5 and .8, where Cronbach's alpha coefficient is regarded as a reasonable indicator of the internal consistency of an instrument and is an appropriate reliability estimate for questionnaires using rating or Likert scales.

In line with the Yale group approach, Messick's (1995) 'unified validity framework' was used to validate the TTKM. This framework treats traditionally separate forms of validity as aspects of a more comprehensive type of construct validity involving content, substantive, structural, generalisable, external and consequential aspects of validity. The internal consistency and validity of the TTKM was established as part of an online survey of software development teams in small to medium enterprises (SMEs) in Ireland the United Kingdom.

6. The Study Procedure

The TTKM was developed through a series of three studies, using a mixed method, sequential exploratory approach (Cresswell, 2003). The studies were as follows:

1. Study 1: Initial scale development using the repertory grid.
2. Study 2: Validation of expert's knowledge by comparison with novices.
3. Study 3: Validation of the TTKM

Each study is now described in detail with concomitant data analysis and results. The results of the three studies are discussed together in section 8.

6.1 Study 1: Initial Scale Development using the Repertory Grid

In this study unstructured interviews were conducted with three experts in the software domain and the context or situation for experts to construe for the repertory grid was decided. This context was in the form of the following question: *What situational factors do you feel significantly influence team performance on successful projects*? This context was chosen because it is general rather than specific allowing respondents' scope to provide their own constructs and was based on the context provided by Moynihan (2002) in his study exploring risk factors in software development projects.

6.1.1 Method

Participants: Thirteen experienced project managers in the software development field from seven different organisations in Ireland (N=6) and the UK (N=1) were chosen as experts. Most managers (N=8) were in the 31-40 age bracket, and from an organisation size of 1000+ (N=5). The mean number of years of experience was 9.23 years (SD = 3.75) and the sample consisted of ten males and three females.

Procedure: We decided to develop a web-based online repertory grid to allow for interaction of previous items in a scale to form part of later items in the questionnaire, a feature necessary to this technique. Online data gathering reduces experimenter bias, can reach a wide geographic audience and was appropriate for the sample which uses computers in their everyday work. The respondents were e-mailed the repertory grid task. The first screen detailed the study and ensured anonymity, participants then completed the online repertory grid.

Step 1: Participants were asked to list five software development projects they had managed. The project managers assigned pseudonyms or aliases to the projects and listed the chosen projects under the following project categories: two 'most successful', one 'in-between' and the two 'least successful' projects they had managed. These project categories were provided in order that comparisons could be made across project managers. Only five projects were chosen because, in a previous study by Moynihan (2002) it was found to be the average number of projects managers listed when no limits were applied. These projects correspond to the elements of the repertory grid.

Step 2: The project managers were asked to think about the five projects in relation to situational factors that affect team performance. The elements (projects) were then presented in groups of three (triads) and the manager was asked to choose the two projects that were similar and thereby different from the third. The respondent was asked to enter a description/term which characterised the way in which the two projects were similar in terms of situational factors that affect team performance. They were then asked to enter the opposite description/term to describe how the third project was different from the other two. This step was repeated 10 times without recurrence of any combination of triads and 10 bipolar constructs were elicited.

Step 3: Each of the five elements (projects) was rated on each of the 10 bipolar constructs on a 1-5 scale where 1 represented the left pole of the construct and 5 represented the right pole. An example of a completed grid is illustrated in Figure 1.

Your Chosen Project Aliases						
	Project A	**Project B**	**Project C**	**Project D**	**Project E**	
Description term	***Most successful (A)***	***Most successful (B)***	***In between***	***Least successful (A)***	***Least successful (B)***	**Opposite description term**
clear goals, close knit team	1	1	5	3	4	varied goals across team
young team-mentoring needed	4	5	5	1	1	experienced team-self determining
multifunctional, experienced	1	1	1	5	5	unifunctional, inexperienced
capable team	2	1	1	3	5	inexperienced
individual performance required	1	5	1	1	1	Team cohesion important
inexperienced manager	5	3	5	1	1	experienced manager
clear team communication	1	1	2	3	5	misinterpreted communication
outsourced remote work	3	5	1	1	5	Single location team
cross-functional team	1	1	1	3	5	discrete team members
clear goals	1	1	2	1	5	unclear goals

Figure 1. Elements rated on each of the 10 bipolar constructs

6.1.2 Results

The results yielded from the online repertory grids were then categorised using content analysis. The thirteen repertory grids yielded 132 bipolar constructs (one manager had two extra constructs) which were numbered, categorised and classified into 27 categories.

In order to counteract the obvious subjectivity of such a categorization a second rater also classified the 132 constructs. There was an inter-rater reliability of 84.84%, with disagreement occurring on 20 constructs which were debated and consensually agreed. Thirteen project managers demonstrated diminishing returns after the 9th elicitation session (manager) when no more new themes emerged. The 27 themes and the number of managers having a construct under each theme can be seen in Table 1.

Theme No.	Theme description	No of managers per theme	No. of constructs per theme
1	Clear well-defined goals	7	21
2	Team is motivated and capable	6	7
3	Co-operative team	5	7
4	Knowledge required for project is available within the team	5	5
5	Clear procedures	4	6
6	Innovative project	4	7
7	Project length	4	4
8	Experienced team	4	4
9	Adequate resources	3	5
10	Diverse team membership	3	5
11	Project scope and importance	3	5
12	Strict deadlines	3	4
13	Third party is involved in the project	3	4
14	Team size	3	4
15	Clearly specified client requirements	3	3
16	Managerial experience and control	3	3
17	Management back up and support	2	6
18	Morale	2	3
19	On schedule and On budget	2	4
20	Measure of Success Criteria in evidence	2	3
21	Clear team communication	2	3
22	Team challenges to management are welcome	2	2
23	Competition within the team	2	2
24	Clear non-competing roles	2	2
25	Client's needs met	1	4
26	Client from same organisation	1	2
27	Other	3	7
	Total		132

Table 1. Number of constructs under each theme

In order to develop a team measure for tacit knowledge regarding the factors that influence team performance on successful software projects, a 'supplied' or 'standardized' grid (Fransella et al., 2004) was developed based on the 27 themes. It was decided to maintain as far as possible the integrity of the managers' original constructs by representing each theme with a verbatim construct, so for example 'Co-operative team' was represented by the construct 'High co-operation'. In all, 25 themes were included in the supplied grid.

6.2 Study 2: Validation of experts' knowledge by comparison with novices

In order to ascertain which of the constructs elicited were truly expert and not mere common sense a comparison was made between novice and expert construing in the second study.

6.2.1 Method

Participants: Ten experts who had already taken part and a further eight reputable project managers were recruited through snowball sampling. Fourteen males and four females with a modal age of 31 – 40 years and average experience of 9.44 years (SD = 3.41), completed the supplied grid. The grid was also administered to 124 final year students in Computer Science at three different Third-level institutions. The modal age was 18 – 24 years with 31 females and 93 males completing the questionnaire. Students were chosen to provide a baseline measure of 'novices'.

Procedure: The supplied grid questionnaire was deployed online and emailed to the experts and novices. The 25 items were answered on a 5-point semantic differential type scale. An example of one of the bipolar constructs was 'Innovative project <----> Mundane/Everyday type project'. Respondents rated the constructs by selecting closest to the statement pole they felt described the factors that influence team performance on successful projects. The questionnaire included 3 demographic questions, referring to age, sex and years of experience.

6.2.2 Results

The novice data had a skewed distribution for most factors. Taking into account the non-normal student sample and the unequal sample sizes between the novices and experts, it was decided to use a non-parametric Mann-Whitney U test to determine which factors differentiated between the two groups Overall et al., 1995). The results of the Mann-Whitney U test revealed 14 items that significantly differentiated between the novices and experts are shown in Table 2. These 14 significant items formed the TTKM.

Theme No.	Left Pole of Bipolar Construct *(scale value of left pole = 1)*	Novice		Expert		Mann Whitney *U*
		Md	*IQR*	*Md*	*IQR*	
1	Clear goals	1	(1 -1)	1	(1 - 1)	864.00*
2	Highly motivated team	1	(2 - 1)	2	(2 - 1)	799.00*
3	Highly co-operative team	1	(1 - 1)	2	(2 - 1)	682.50**
4	Knowledge required available within the team	2	(3 -1)	3	(3 - 2)	708.00**
5	Unclear Procedures	5	(5 - 4)	4	(5 - 3)	1053.00
6	Innovative project	2	(2 - 1)	3	(4 - 2)	443.50***
7	Short-term project	3	(3 - 2)	3	(3 - 2)	952.50
8	Experienced team	2	(2 - 1)	2	(3 - 2)	777.00*
9	Adequate resources	1	(1.75 - 1)	2	(3 - 1)	544.00***
10	Diverse team membership	2	(3 - 2)	3	(3 - 3)	640.00**
11	Small project	3	(3 - 2)	2	(3 - 2)	752.00*
12	Strict deadlines	2	(3 - 2)	3	(3 - 2)	742.50*
13	Third party is involved in the project	3	(4 - 2)	3	(4 - 3)	856.00
14	Big Team	4	(4 - 3)	4	(3 - 1)	762.00*
15	Inaccurate client requirements	5	(5 - 4)	5	(5 - 3.75)	1015.00
16	Manager in control	2	(2 - 1)	2	(3 - 1)	1115.00
17	Management back up and support	2	(2 - 1)	2	(2 - 1)	988.50
18	Low morale	5	(5 - 4)	4	(5 - 4)	795.50*
19	On schedule and On budget	2	(3 - 1)	2	(3 - 1)	1078.50
20	Lack of measures of success	4	(5 - 3.25)	4	(5 - 3.75)	1048.00
21	Clear team communication	1	(1 - 1)	1	(2 - 1)	929.00
22	Management decisions challenged	2	(2 - 1)	2.5	(3 - 1)	989.50
23	Internal competition	3	(3 - 2)	4	(5 - 3)	556.50***
24	Clear non-competing roles	1	(2 - 1)	1	(3 - 1)	1085.50
25	One clearly identified decision maker/leader	2	(3 - 1)	1	(2 - 1)	628.00**

*p<.05; **p<.01; ***p<.001

Table 2. Medians (Md), interquartile (IQR) range and Mann Whitney U for novices and experts

The 18 expert responses for each of the 14 bipolar constructs were used to create an expert profile using the expert response means. The resultant TTKM was a 14 item bipolar, situational judgement test where participants indicated the degree to which they felt each of 14 factors affected team performance on successful software development projects and responded on a 5-point semantic differential type scale. The 14 constructs were rated by selecting closest to the statement pole that best describes the factors that influence team performance on successful projects. The TTKM constructs and expert profile are shown in Table 3.

	Left Pole of Construct *(Scale value of left pole = 1)*	Mean	SD	Right Pole of Construct *(Scale value of right pole =5)*
1	Clear goals	1.00	0.00	Vague goals
2	Highly motivated team	1.56	0.51	Team is not motivated
3	Highly co-operative team	1.72	0.75	Uncooperative team
4	Knowledge required available within the team	2.56	0.92	Knowledge required is not
5	Innovative project	3.11	0.96	Totally available within the team
6	Experienced team	2.16	0.62	Mundane/everyday type project
7	Adequate resources	2.00	0.77	Inexperienced team
8	Diverse team membership	2.89	0.83	Inadequate resources
9	Small project	2.44	0.51	Uniform team membership;
10	Strict deadlines	2.72	0.46	Extensive project
11	Big Team	4.17	0.78	Widely variable deadlines
12	Low morale	4.28	0.46	Small team
13	Internal competition	3.83	1.09	High morale
14	One clearly identified decision maker/leader	1.44	0.70	No competition within the team

Table 3. Expert means and standard deviations for the 14 TTKM bipolar constructs

6.3 Study 3: Validation of the TTKM

The final study was conducted to establish the internal consistency and validity of the TTKM. The TTKM was incorporated into a larger online survey for software development teams. The survey included an index of the quality of social interaction (QSI) within the team, team performance as measured by efficiency and effectiveness and two items measuring explicit knowledge and other demographic variables.

Participants: Forty eight teams in 46 SMEs from Ireland and the UK completed the questionnaire consisting of 75% (N=121) males and 25% (N=60) females. Most (47%) were in the 31-40 age range with an average of 11.64 years of experience (SD 4.97).

6.3.1 Team Size and Within-Team Response Rate

In this study, respondents listed the initials of up to a maximum of 11 team members excluding the respondent. Team size varied from 2 to 12+, with the mean team size being 4.91 and an average within team response rate of 81.86% which was deemed acceptable.

6.3.2 Measures and Scoring

The TTKM was scored by comparing the individual score on each of the 14 bipolar constructs items with an expert profile. We scored the responses to the TTKM by calculating a squared Euclidean distance of the individual from that of the expert mean. These individual scores were then aggregated to form a team score, and the r_{wg} for the TTKM scale was .96, indicating homogeneity and that aggregating members' scores to the team-level of analysis was statistically justified.

Quality of Social Interaction (QSI): The quality of social interaction was assessed by a self-report questionnaire regarding two perceived outcomes of social interactions across team members, resulting in an index of social interaction. This measure was adapted from Chiu et al., (1995) in which participants were asked to recall the most recent instance where they spent more than 15 minutes alone interacting face-to-face with each member of the team. For each team member participants were asked to (a) indicate on a 4 point scale whether they had attained their goal in the interaction (where 0 = ‘ not applicable’, 1 = ‘no’, 2 = ‘to some extent’ and 3 = ‘yes’), and (b) indicate the degree of change in their relationship with the other person after the interaction, also on a 4 point scale where 0 = ‘not applicable, 1 = ‘got worse’, 2 = ‘remained the same’ and 3 = ‘got better’. Therefore the possible range was from 0 – 9. In line with Chiu et al.’s (1995) analysis for each interaction, the responses to the two questions were multiplied to form an interaction quality index for that social interaction. All of the interaction quality indexes were averaged to form an overall index of perceived interaction quality for each individual. These scores were then aggregated to form a team score of quality of social interaction.

Team Performance: Two dimensions for performance, Effectiveness and Efficiency were measured (Faraj and Sproull, 2000). Self report measures were chosen because objective measures of performance present difficulties in the Information Systems field (Henderson and Lee, 1992) since ‘using objective measures assumes comparability across software projects or unique situations constraints, and this raises a new set of methodological measurement issues’ (Faraj and Sproull, 2000, p.1560).

The effectiveness measure consisted of five items and asked how well teams performed, in relation to other software development teams they have known, on dimensions of work quality, team operations, ability to meet project goals, extent of meeting design objectives and reputation of work excellence. The efficiency measure had two items and dealt with adherence to schedule and budget. Responses for both effectiveness and efficiency were rated on a 1 to 5 scale from ‘not very good’ to ‘excellent’. The r_{wg} for the effectiveness scale was .90 and .76 for efficiency.

Explicit Job Knowledge: Two self report items were developed to measure perceived explicit knowledge which was operationalised as official job knowledge. Explicit knowledge was assessed by asking respondents their perceived levels of familiarity with official written procedures rated on a 5-point scale from 1 = ‘not at all familiar’ to 5 = ‘very familiar’ and their degree of reliance on official written procedures involved in carrying out their work also rated on a 5-point scale from 1 = ‘I mostly rely on written procedures’ to 5 = ‘I never rely of written procedures’ (reverse scored). Individual scores were averaged to team-level.

6.3.3 Internal Consistency of the TTKM

The internal consistency for all measures at individual level are all above alpha = 0.68 (except for the TTKM) and above alpha = 0.67 at the team level. Therefore, the internal consistency of the measures is considered adequate at the team level. Internal consistency for the TTKM as measured by Cronbach’s coefficient alpha was a=.49 at the individual level and a=.71 at the team-level. This indicates a significant increase in the internal reliability of the measure at the team-level, thus providing support for the premise that TTKM measures tacit knowledge at the team rather than individual level. Given that the obtained team-level reliability falls within the range for other situational judgement tests and for those reliabilities obtained on previous measures of tacit knowledge then we consider the internal consistency of the team-level score to be acceptable.

6.4 Results

First individual scores were calculated for all variables which were then averaged for team-level analysis. Table 4 presents the means, standard deviations, and bivariate correlations among the study variables and the internal consistency of the measures.

Variable	M	SD	1	2	3	4	5
1. TTKM	5.49	2.48	(.71)				
2. QSI	12.83	1.88	.45**	(NA)			
3. Effectiveness	3.69	0.55	.35*	.13	(.88)	.	
4. Efficiency	3.24	0.73	.09	-.03	.56**	(.83)	
5. Reliance on written procedures	3.00	0.95	.20	.20	.20	.15	(NA)
6. Familiarity with written procedures	4.07	0.77	.19	.18	.30*	.33*	.48** (NA)

NB. Values in parentheses represent internal consistency reliabilities (Cronbach's alphas) obtained in this study
*p<.05, **p<.01

Table 4. Means, standard deviations, reliabilities and inter-correlations among study variables (N = 48)

6.4.1 Establishing Validity

Using Messick's (1995) unified validity framework the content relevance and theoretical rationale for the TTKM were established by domain experts, in this case expert software project managers in order to determine boundaries of the construct domain. The TTKM items reflected knowledge acquired through practical experience not formal instruction. Selecting items that differentiated novices and experts provided further validation for the TTKM since the structural aspect of the measure was formed from the result of the expert-novice response differences on the items. Generalisation to others in the domain was established using test-retest reliabilities for repeated administration of the test across individuals. Furthermore, the survey was piloted on two software teams where no negative consequences associated with items of the TTKM were uncovered.

The external aspect of construct validity refers to convergent and discriminant validity. Scores on the TTKM had a low non-significant relationship with explicit job knowledge as measured by 'reliance on written procedures' ($r = .20$, $p>.05$) and 'familiarity with written procedures ($r = .19$, $p>.05$), discriminating the construct from explicit job knowledge. Convergent validity was provided by a significant correlation between scores on the TTKM and scores on the QSI ($r = .45$, $p<.01$) providing some empirical support for the theoretical argument, that tacit knowledge is diffused and acquired through social interaction and supporting. In terms of predictive validity the TTKM was significantly related to the effectiveness component of team performance ($r = .35$, $p<.05$) but not the efficiency aspect, with effectiveness and efficiency correlating well together ($r =.55$, $p<.01$).

Tables 5 and 6 represent the hierarchical regression analyses performed to examine if scores on the TTKM predicted team performance (effectiveness and efficiency respectively) over and above quality of social interaction and explicit job knowledge. It was found that overall the model did not significantly predict either effectiveness or efficiency, however, scores on the TTKM were significant in predicting effectiveness accounting for 8% of the variance over and above social interaction and explicit job knowledge. In addition familiarity with written procedures was significant in predicting efficiency. The two significant results are further illustrated in Figures 2 and 3 respectively.

	Step 1		Step 2						
Variables	β	t	β	t	Df	R^2	F	ΔR^2	ΔF
Step 1: Control Variables									
QSI	.07	0.51	-0.6	-0.38	3, 44	.10	1.58		
Familiarity with written procedures	.25	1.53	.23	1.43					
Reliance on written procedures	.06	0.39	.04	0.23					
Step 2: Tacit Knowledge									
TTKM			.32	2.05*	4, 43	.18	2.32	.08	4.21*

* p<.05

Table 5. Summary of hierarchical regression analysis for variables predicting effectiveness (N = 48)

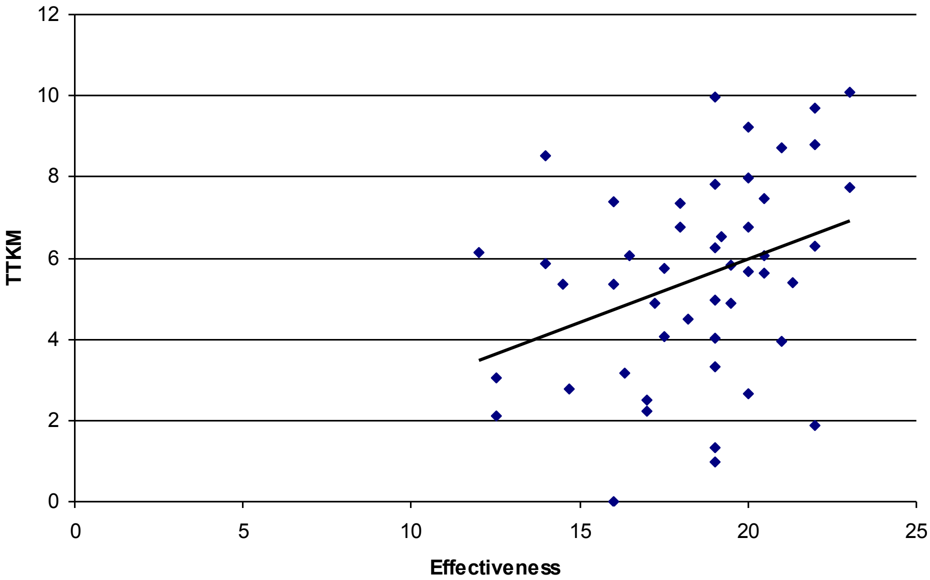

Figure 2. Regression for predicting effectiveness

	Step 1		Step 2						
Variables	β	t	β	t	Df	R^2	F	ΔR^2	ΔF
Step 1: Control Variables									
QSI	-.10	-.66	-.13	-0.83	3, 44	.12	1.93		
Familiarity with written procedures	.34	2.09	.33	2.03*					
Reliance on written procedures	.10	0.60	.00	0.01					
Step 2: Tacit Knowledge									
TTKM			.09	0.56	4, 43	.12	1.50	.00	0.58

* p<.05

Table 6. Summary of hierarchical regression analysis for variables predicting efficiency (N = 48)

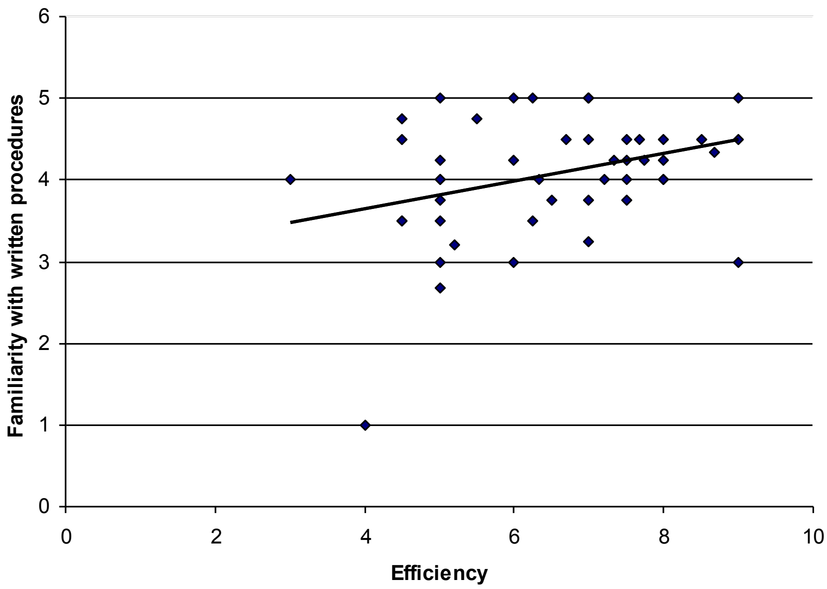

Figure 3. Regression for predicting efficiency

7. Discussion

The development of the TTKM for software development teams is an extension of individual-level, tacit-knowledge research to consider team-level behaviour. The repertory grid technique was used to develop the scale items and it was found to be effective in tapping the tacit knowledge construct at the articulated level of abstraction, since after content analysis, constructs that differentiated novices from experts were established leading to the final TTKM. Using Messick's (1995) unified validity framework, it was found that the TTKM was a reasonably internally valid and reliable measure of tacit knowledge at the team-level, but not at the individual level, though more studies would be needed to further test the reliability and expand the validity to other populations.

In addition to the development of the TTKM, we posed three questions in section 1 of this paper. The following sub-sections will now answer these in light of the results obtained in the development of the TTKM.

7.1 Does team tacit knowledge predict performance in software development teams?

In terms of predictive validity the TTKM was significantly related to the effectiveness component of team performance but not the efficiency aspect. In addition, scores on the TTKM significantly predicted effectiveness over and above the other factors in this study illustrating the importance of team tacit knowledge in influencing this aspect of team performance. These findings were consistent with the nature of efficiency and effectiveness in teams. Efficiency relates to budgeting and scheduling and has been found to be associated with formal administrative co-ordination and reporting procedures which themselves have not been found to be related significantly to effectiveness (Faraj and Sproull, 2000).

Effectiveness on the other hand is characterised by how well the team meets project goals, the quality aspect rather than speed and budget. Theoretical support for this finding links tacit knowledge to competitive advantage (Fernie et al., 2003). However, it appears that tacit knowledge only contributes partly to competitiveness since effectiveness refers to the achievement of project goals and not budget and schedule. Much of the evidence linking tacit knowledge to team performance is anecdotal (e.g., Nonaka and Takeuchi, 1995). Empirical evidence for tacit knowledge and team performance (effectiveness) at the individual level was demonstrated by the Yale group studies, (e.g., Sternberg and Wagner, 1988) and at the team-level, by Berman et al. (2002). This finding is also in accordance with Guinan et al. (1998) who found that processes focussing on the team's relationships may be more likely than technical factors to be associated with team performance on successful projects.

7.2 Is there a relationship between tacit knowledge and explicit job knowledge?

There was low non-significant relationship between tacit knowledge and explicit job knowledge as measured by 'reliance on written procedures' and 'familiarity with written procedures', thus providing discriminant validity for the TTKM. This finding concurs with theoretical propositions defining tacit knowledge in that it is acquired with little environmental support and not through formal means (Sternberg et al., 2000). This implies that tacit knowledge regarding the factors that affect team performance on successful projects is not written down or formalised in work practices, but is altogether more practical and experience based.

7.3 Do teams with higher levels of social interaction also have higher levels of tacit knowledge?

There was a significant correlation between scores on the TTKM and quality of social interaction ($N = 48$, $r = 0.45$, $p<.01$) thus demonstrating that team tacit knowledge is positively related to quality of social interaction, since the TTKM was not significantly related to quantity of social interaction. This indicates that the nature and quality of the informal social interactions are related to the amount of tacit knowledge within a team, while the frequency of interaction is not. Tacit knowledge and quality of social interaction were defined as being tied to the achievement of goals and the theoretical and mainly anecdotal link between these concepts was established empirically in this study. This finding is supported by the theoretical argument posited by Nonaka and Taakeuchi (1995) and SNA study by Busch et al. (2003) who concluded that tacit knowledge is shared through social interaction.

8. Conclusions

Theoretical claims about the tacit knowledge construct are abundant, however, there is a paucity of empirical studies, due in part to the lack of empirical measures, and in part, the problems of conceptualising and defining tacit knowledge. Three conclusions are forwarded in relation to the tacit knowledge construct.

1. It may be concluded that, tacit knowledge can be measured at the articulated level of abstraction. The development of the TTKM for software development teams, is an extension of individual-level, tacit-knowledge research to consider team-level behaviour.
2. It may also be concluded that the TTKM, is a valid and reliable measure of tacit knowledge at the team level.
3. It may be concluded that, tacit knowledge is not job knowledge, since tacit knowledge at the team level is not related to explicit job knowledge.

In relation to how tacit knowledge is acquired and shared in software development teams, the present study advances our understanding of this process and the following conclusions are made:

1. Tacit knowledge is acquired and shared directly, through good quality social interactions.
2. The frequency of interaction indirectly aids the acquisition and sharing of tacit knowledge since it leads to better quality interactions.

A question posed in the present study is the predictive capacity of team tacit knowledge for team performance as measured by effectiveness and efficiency. The following conclusions are forwarded:

1. Team tacit knowledge is an important factor in the prediction of effectiveness but not efficiency. Team tacit knowledge does predict effectiveness above and beyond quality and quantity of social interaction. Therefore, team tacit knowledge and the coordination of specialised knowledge within teams are significant factors in effective performance for software development teams.
2. Efficiency in software development teams, is related to explicit, formal procedures i.e. presence of a formal knowledge sharing and administrative coordination. Efficiency was also found to be related to explicit job knowledge (familiarity with written procedures) and presence of expertise. It is concluded that efficiency is generally associated with explicit knowledge and formal procedures while effectiveness is predicted from tacit knowledge and non-formal procedures.
3. A further conclusion is that quality and quantity of social interaction are not directly related to team performance.

The practical implications for knowledge software development teams are :

1. Team tacit knowledge can explain how members of effective software development teams apply what they know.
2. The team tacit knowledge construct can help us differentiate between low- and high-performing teams by suggesting that members of high-performing teams have developed different aspects of tacit knowledge about successful performance on projects and this knowledge is then applied to team tasks which can be seen in performance.
3. Software development teams work with intangible cognitive processes and are knowledge-workers, where expertise in software development teams requires coordination.

8.1 Implications

The practical implications of this research for software development teams may be that the tacit knowledge construct can allow us to differentiate between low and high-performing teams by suggesting that members of high-performing teams have developed different aspects of tacit knowledge about successful performance on projects and this knowledge is then applied to team tasks which can be seen in effective performance. Managers of software development teams can make changes within the organisation and to the team to enhance social interaction and encourage the sharing of tacit knowledge thereby increasing team effectiveness. Firm conclusions cannot be drawn as to how managers go about increasing social interaction as it was not addressed in this study.

The implications for software development teams are that processes and methods that may encourage interaction, may also lead directly and indirectly to an increase in the tacit knowledge base. The outcomes of this study may provide some theoretical background and empirical evidence for the use of Agile methods for software development. Proponents of Agile Methods such as Chau and Maurer (2004) posit that there is a need for knowledge sharing to enable software organisations to leverage tacit knowledge and that this knowledge sharing would occur through face-to-face interactions. The present study has demonstrated that tacit knowledge is related to quality of social interaction. These issues are accounted for by Agile methods and it appears that the Agile approach to software development, by extrapolation may enhance effectiveness.

As software project failure is a serious problem for the software development industry, investigations into improving tacit knowledge acquisition in teams may influence successful project outcomes. Finally, the technique to develop the TTKM may be used to measure team-level tacit knowledge in any domain not just software development expanding the possibilities for uncovering and comparing similarities and differences in team-level tacit knowledge.

8.1 Limitations and Future Directions

In addition to the usual issues associated with field research other limitations are now identified. Firstly, delivering the repertory grid technique online did allow for speed and cut across geographical boundaries however face-to-face administration may provide deeper and richer knowledge. The trade-off between the two was deemed necessary and practical since that time demands were kept to a minimum for each participant.

The participants in the first study were not randomly chosen, instead snowball sampling was used which introduces bias in the selection of respondents. This bias was a necessary part of the expert sample since the study sought expert project managers who had a reputation for excellence. A further limitation of this study is that there is no way of knowing if the team members collaborated or interacted with one another while completing the questionnaire. However, the existence of standard deviations across responses, on all measures in all teams provides some support that the teams did not collaborate. Research with larger samples and varied team types is needed to validate the TTKM scale and examine how the TTKM differs among teams. The TTKM is not intended to be a comprehensive measure of tacit knowledge. Future measures employing our technique could be developed to measure different aspects of tacit knowledge in software development teams to include a detailed approach to processes or languages used while developing software. Team performance was measured using self-report, it may be useful in future to triangulate, this measure with other forms of effectiveness and efficiency assessment, e.g. stakeholder measures or lines of code.

The development of the TTKM formed part of a larger study investigating the acquisition and sharing of tacit knowledge in software development teams where it was proposed that teams acquire and share tacit knowledge through social interaction and the development of a specific team mental model called a transactive memory system.

References

Baumard, P., 1999. Tacit Knowledge in Organizations, Sage, London.
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.
Blackler, F., 1995. Knowledge, knowledge work and organizations: An overview and interpretation, Organization Studies 16(6), 1021-1046.
Boehm, B., Turner, R., 2004, Balancing Agility and Discipline, Addison Wesley
Busch, P., Richards D., Dampney, C. N. G., 2003. The graphical interpretation of plausible tacit knowledge flows, Conferences in Research and Practice in Information Technology, 24, Australian Computer Society.
Cannon-Bowers, J. A., Salas, E., Converse, S. A., 1993. Shared mental models in expert team decision making. In: N. J. Castellan (Ed.), Individual and Group Decision Making, Erlbaum, Hillsdale, New Jersey.
Chau, T., Maurer F., 2004. Knowledge Sharing in Agile Software Teams. In: Proceedings of the Symposium on Logic versus Approximation, Springer Verlag
Chiu, C., Hong, Y, Mischel, W., Shoda, Y., 1995. Discriminative facility in social competence: Conditional versus dispositional encoding and monitoring-blunting of information, Social Cognition, 1(1), 49-70.
Cockburn, A., Highsmith, J., 2001. Agile Software Development: The People Factor, IEEE Computer, Vol. 34, No.11, 2001.
Creswell, J. W., 2003. Research Design: Qualitative, Quantitative, and Mixed Methods Approaches, Second ed., Sage, London.
DeMarco, T., Lister, T., 1987. Peopleware: Productive Projects and Teams, Dorset House, New York.
Desouza, K. C., 2003. Facilitating Tacit Knowledge Exchange, Communications of the ACM, 46(6), 85-88.
Drucker, P., 1993. Post-Capitalist Society, Harper Business, New York.
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, Decision Sciences, 34(2), 197-223.
Faraj, S., Sproull, L., 2000. Coordinating expertise in software development teams, Management Science, 46 (12), 1554-1568.
Fernie, S., Weller, S., Green, S.D., Newcombe, R., 2003. Knowledge sharing: context, confusion and controversy, International Journal of Project Management, 21(3), 177-187.
Frangos, A. 2004, Water-Cooler Conversations Escalate to Office Stairwells, The Wall Street Journal Online, Available from www.careerjournal.com
Fransella, F., Bell, R., and Bannister, D., 2004. A Manual for the Repertory Grid Technique. Wiley.
Gottfredson, L.S., 2003. On Sternberg's "reply to Gottfredson", Intelligence, 31, 415-424.
Gourlay, S. N., 2006. Towards conceptual clarity concerning "tacit knowledge": a review of empirical studies, Knowledge Management Research and Practice 4(1), 60-69.
Grant, R. M., 1996. Toward a knowledge-based theory of the firm, Strategic Management Journal, 17(Special Issue), 109-122.
Guinan, P., Cooprider, J., Faraj, S., 1998.Enabling software development team performance during requirements definition: A behavioral versus technical approach, Information Systems Research, 9(2), 101-125.
Hedlund J., Forsythe G. B., Horvath, J. A., Williams, W. M., Snook, S., Sternberg R.J., 2003. Identifying and assessing tacit knowledge: understanding the practical intelligence of military leaders, Leadership Quarterly, 14(2), 117-140
Henderson, J., Lee, S., 1992. Managing I/S design teams: A control theories perspective. Management Science, 38(6), 757-777.
Hutchins, E., 1991. The social organization of distributed cognition. In: L. B. Resnick, J. M. Levine and S. D. Teasley (Eds.), Perspectives on Socially Shared Cognition, American Psychological Association, Washington DC.
James, L. R., Demaree, R .G., Wolf, G., 1984. Estimating within-group interrater reliability with and without response bias, Journal of Applied Psychology, 69(1), 85-98.
Jha, S. R., 2002. Reconsidering Michael Polanyi's Philosophy. University of Pittsburgh Press

Kelly, G. A. (1955/1991). The Psychology of Personal Constructs (Vols. 1 & 2), Routledge.
Klein, K. J., Dansereau, F., Hall, R. J., 1994. Levels issues in theory development, data collection, and analysis, Academy of Management Review, 19, 195-229.
Legree, P., 1995. Evidence for an oblique social intelligence factor established with Likert-based testing procedure, Intelligence, 21, 247-266.
Lewis, K., 2003. Measuring transactive memory systems in the field: Scale development and validation, Journal of Applied Psychology, 88(4), 587-604.
Messick, S., 1995. Validity of psychological assessment: Validation of inferences from persons' responses and performances as scientific inquiry into score meaning, American Psychologist, 50, 741-750.
Mohammed, S., Dumville, B. C., 2001. Team mental models in a team knowledge framework, Journal of Organizational Behavior, 22, 89-106.
Moynihan, T., 2002. Coping with IS/IT Risk Management: The Recipes of Experienced Project Managers, Springer-Verlag, London.
Nonaka, I Takeuchi, H., 1995. The Knowledge Creating Company, Oxford University Press, New York.
Overall, J. E., Atlas, R. S., Gibson, J. M., 1995. Tests that are robust against variance heterogeneity in k x 2 designs with unequal cell frequencies, Psychological Reports, 76, 1011-1017.
Polanyi, M., 1966. The Tacit Dimension, Routledge, London.
Reber, A. S., 1995. Implicit Learning and Tacit Knowledge: An Essay on the Cognitive Unconscious, Oxford University Press, New York.
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.
Royal Academy of Engineering, 2004, The Challenges of complex IT Projects, Available from www.bcs.org/statements/royal
Standish Group International, 2004. CHAOS Demographics-2004 Third Quarter Research Report.
Sternberg, R. J., Forsythe, G. B., Hedlund, J., Horvath, J., Wagner, R. K., Williams, W. M., Snook, S. A., Grigorenko, E. L., 2000. Practical Intelligence in Everyday Life, Cambridge University Press.
Sternberg, R. J., Wagner, R. K., 1988. Practical Intelligence: Nature and Origins of Competence in the Everyday Word, Cambridge University Press, New York.
Stewart V. Stewart, A., 1981. Business Applications of the Repertory Grid, McGraw Hill, London.
Taub, G. E., Hayes, B. G., Cunningham, W. R., Sivo, S. A., 2001. Relative roles of cognitive ability and practical intelligence in the prediction of success, Psychological Reports, 88, 931-942.
Von Krogh, G., Roos, J., 1995. A perspective on knowledge, competence and strategy, Personnel Review, 24(3), pp. 56-76
Wagner R.K., Sternberg R. J., 1985. Practical intelligence in real-world pursuits: The role of tacit knowledge, Journal of Personality and Social Psychology, 49(2), 436-458.
Wagner R.K., Sternberg R. J., 1991. TKIM: The Common Sense Manager: Tacit Knowledge Inventory for Managers: Test Booklet, Harcourt Brace Jovanovich, San Antonio.
Weick, K. E., 1979. The Social Psychology of Organizing, second ed., Addison-Wesley, Reading, MA.