Computer Systems and Software Engineering:

Concepts, Methodologies, Tools, and Applications

Information Resources Management Association
USA

Volume I

IGI Global
DISSEMINATOR OF KNOWLEDGE

Published in the United States of America by
IGI Global
Engineering Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue
Hershey PA, USA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com

Library of Congress Cataloging-in-Publication Data

Names: Information Resources Management Association, editor.
Title: Computer systems and software engineering : concepts, methodologies, tools, and applications / Information Resources Management Association, editor.
Other titles: Computer systems and software engineering (Hershey, Pa.)
Description: Hershey, PA : Engineering Science Reference, [2018] | Includes bibliographical references.
Identifiers: LCCN 2017026268| ISBN 9781522539230 (hardcover) | ISBN 9781522539247 (ebook)
Subjects: LCSH: Computer systems. | Software engineering.
Classification: LCC QA76 .C5736 2018 | DDC 004--dc23 LC record available at https://lccn.loc.gov/2017026268

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

Chapter 19
Understanding the Role of Knowledge Management in Software Development:
A Case Study in Very Small Companies

Rory V. O'Connor
School of Computing, Dublin City University, Ireland

Shuib Basri
University of Limerick, Ireland & Universiti Teknologi Petronas, Malaysia

ABSTRACT

Software and systems engineering is distinct from other forms of engineering as it deals with an intangible product, where the progress in construction is not explicitly visible and team members often rely on the documentation of others to follow and review progress. Furthermore unlike traditional engineering disciplines, there is no single standardized unified process. The role of knowledge management in the software engineering literature is becoming more evident, as the software development activity is essentially a human knowledge intensive activity and is seen by many as a key factor. This paper discusses the role of software development knowledge management within software development process and specifically how software development knowledge is managed in software development in order to support software process improvement and the role of knowledge management in this. The authors present the results of a study of knowledge management process practices in very small software companies and discusses these under the major identified issues of: Communication; Learning and sharing; Documentation and Knowledge management process and commitment. The findings in this study give an insight towards knowledge management practices as they relate to software development process practices in very small companies and the important factors that must be considered to preserve knowledge and quality software.

1. INTRODUCTION

The majority of small and very small software companies find implementing controls and structures to properly manage their software development activity to be a major challenge. The administration of the software development activity is usually achieved through the execution of a software development

DOI: 10.4018/978-1-5225-3923-0.ch019

process, which describes the way an organization develops its software products and supporting services, such as documentation. Such software processes define what steps the development organizations should take at each stage of production and also provide assistance in making estimates, developing plans and measuring quality. The process and associated activities are often documented as sets of procedures to be followed during development, however, the documentation is not the process but should clearly represent the process as it is implemented within an organization.

At the core of all software development activity are the human beings that implement the software development process in order to produce the actual software systems. In this context human beings gain expertise through perception, intuition and experience, rather than by following a predefined process (Dreyfus et al., 1986). In support of this it has been argued argues that software engineering is knowledge study and hence knowledge management is of high importance in software engineering that (Edwards, 2003), which clearly has implications for the management of knowledge in software development. Therefore we are interested in understanding the role of software development knowledge management within software development companies. Specifically out focus is on how software knowledge is managed; identify critical factors in software development teams and software development knowledge management; understand how should software teams are organized in order to support software process improvement and the role of knowledge management in this.

1.1. The Software Development Process

There are multiple approaches to organizing the software development process and multiple factors influencing the software development process (Clarke & O'Connor, 2012), with two major ones being the traditional (or plan based), which rely primarily on managing explicit knowledge, and agile methods, which primarily rely on managing tacit knowledge and recognises the importance of human interaction in the software process over written knowledge in formal documentation. Therefore understanding the role and nature of knowledge in software development is key challenge (Ryan & O'Connor, 2013).

Essentially a software process is all the stages, tasks and activities that are followed by an organization to develop a software product (Zahran, 1998). The software process has four distinct roles; (i) to present a guidance as the guideline of the activities to be undertaken; (ii) to specify the artefact that should be developed and when; (iii) to direct the task of the development team; and (iv) to offer ways of monitoring and measuring a project progress and output (Kruchten, 2000). It is commonly agreed that the software process must be evolved (maintained and improved over time) in order to meet current evolving business needs, thus the area of Software Process Improvement (SPI) has gained increased importance in software engineering area. The aim of SPI is to understand the software process as it is used within an organization and thus drive the implementation of changes to that process to achieve specific goals such as increasing development speed, achieving higher product quality or reducing costs. There is a widely held belief that a better software process results in a better software product, with authors such as Humphrey (1989) claiming that to improve your product, you must improve your process quality. Although there is evidence that many organizations do not sill do not subscribe to the process improvement philosophy despite the widely agreed benefits (O'Connor & Coleman, 2009).

The software development activity is essentially a human knowledge intensive activity, involving software developers executing a software development process utilizing expert knowledge, within a team. Accordingly we are interested in understanding the role of software development knowledge management within software development companies. Specifically: How software knowledge is managed; identify

critical factors in software development teams and software development knowledge management; understand how should software teams are organized in order to support software process improvement and the role of knowledge management in this.

1.2. Knowledge Management

Knowledge Management (KM) is a discipline that crosses many areas such as economics, informatics, psychology and technology. KM is seen as a strategy that creates, acquires, transfers, consolidates, shares and enhances the use of knowledge in order to improve organizational performance and survival in a business environment. This scenario becomes a challenge to the companies in managing their organizational knowledge (Kukko et al., 2008). Therefore specific plans and suitable tools will guide the knowledge management process (Dingsoyr & Conradi, 2002). This plans and tools must be promoted applying the old knowledge to new situations in an organization (Kukko et al., 2008). In software engineering, individuals are the most important actor in KM, who perform tasks for achieving goal that been set by the organizational level. Through social and collaborative work among the people in an organization, process knowledge is created, shared, amplified, enlarged and justified on organizational setting (Nonaka & Takeuchi, 1995). Moreover knowledge is about action-outcome and the effects of the firm environment (Weick, 1995) and was created through a conversion between tacit knowledge and explicit knowledge (Nonaka et al., 2000).

In KM, knowledge creation and sharing is a continuous process whereby individuals and groups within the organization and between the organizations share tacit and explicit knowledge (Ryan and O'Connor, 2009). The organization capability to create knowledge is important in order to sustainable competitive advantage (Nonaka et al., 2000; Parent et al., 2000). Knowledge creation process is believed started when an individual recognize the related and useful data and information and then able to transform it into a new knowledge that brings a future value to an organization. Organizational knowledge is not only created within the organization but also can be acquired externally and this can be done through knowledge sharing (Grant, 1996; Awazu, 2004; Nonaka & Takeuchi, 1995). The important of knowledge sharing and knowledge creation in any organization will help organization to continuously innovate and help organization to sustain their competitiveness (Rhodes et al., 2008). These activities are usually supported by a social network within an organization and through the development between departments in an organization link (Szulanski, 1996). In addition, Turner and Makhija added that in sharing and creating knowledge, trust and organizational control plays an important role in how individual transferring and sharing their knowledge with others in an organization (Turner & Makhija, 2006).

Knowledge is vital for every organization because it is needed to perform a work in an organization. According to Hendricks and Vriens (1999) an organization cannot survive and sustain their competitiveness without knowledge. Therefore knowledge needs to be managed to ensure that the right knowledge gets into the right place. This also will increase the innovation power of organization and its knowledge worker. In addition knowledge in organization also will be eroding over the time and will contribute to loss of knowledge in organization. This condition is often implicit and its loss is often not recognizing until too late. According to Shaw et al. (2003) knowledge erosion is referred as the loss of knowledge resulting from people leaving an organization or changing jobs within it. Several authors claim that knowledge erosion became one of the main problems as the organization expanding over the time (Litern, 2002). The lacking of resource and time in small company in implement knowledge management will introduce a knowledge erosion situation through employee retirement and resignation (Bjorson &

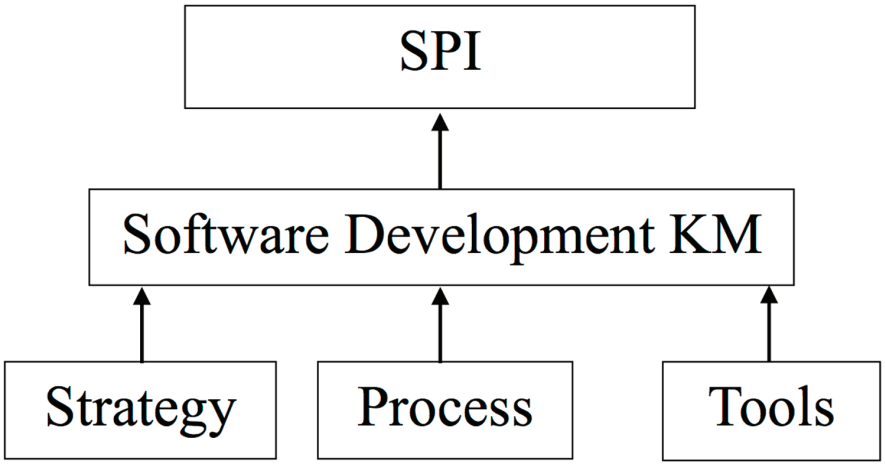
Dingsoyr, 2008). In addition, 4 important criteria in organization; the staff development, team building, communication of role and function, and formal continuous process improvement; was believed could help organization in mitigated this issue (Shaw et al., 2003). Accordingly, the issue of knowlegde loss in software deveopment roganisations, as its pertains to software development peocess is a key isues for many software development organisations (Basri & O'Connor, 2011a).

1.3. Knowledge Managament in SPI

Software process is not standardized in all software projects (Borges & Falbo, 2002). Software process must be updated and improved frequently in order to cope with any environment changes. Such environment required KM in supporting software process definition and activities (Sirvio et al., 2002). Hansen and Kautz (2004) explained that SPI could strengthen KM abilities for software development organization. In term of small organization, Kettunen (2003) argues that KM is core to a software process improvement model and that the relationship between SPI and organizational learning are very strong. They points out that people in an organization will create, acquire and share knowledge continuously in order to improve software development practices. Moreover, in nowadays business environment where software development project becoming more complex, the greater reliance upon the knowledge processes to resolves problems are really important (Aurum et al., 2003; Bjornson & Dingsoyr, 2005). Bjornson and Dingsoyr (2008) stated in their review that proper managing of organizational knowledge is important in SPI efforts and it is a major factor for success. Mathiassen and Pourkomeylian (2003) in their survey on practical usage of KM to support innovation in a software organization claims that KM and SPI are very close related. They added that knowledge management is used to update practices within software organization generally and SPI specifically. According to Sirvio et al. (2002) software organization needs to improve their practices in order to cope with market changes. These situations have lead to considerable interest in how organization can effectively respond to changing environment or agile environment (Aaen et al., 2007; Sirvio et al., 2002). Therefore KM is seen as critical to the SPI process.

Therefore, based on the above discussion we proposed a study model as depicted in Figure 1. From the diagram, it shown that the SPI and software development KM are related to each other. This relation-

Figure 1. The study model



ship is vital in preventing knowledge atrophy and process erosion problems. In addition, a proper KM process could help software teams become more effective in performing team task and making a decision (Kettunen, 2003). (Aaen et al., 2002) added with an appropriate knowledge creation and sharing process could provide team members with clear SPI goals and sustain their interest.

1.4. Very Small Companies

Due to the rich variety of software development settings (for example: the nature of the application being developed, team size, requirements volatility), the implementation of a set of practices for software development may be quite different from one setting to another (Jeners et al., 2013). Small and very small companies are the fundamental growth of many national economies. It is important to notice that the contribution from the small companies should be seen as important and significant as compare to the large one. The majority of software companies are small and for example in Ireland the majority of the Irish indigenous software firms are employed between 10 to 99 employees and average size is about 16 employees (Coleman & O' Connor, 2008a). The same scenario occurs in many other countries especially in Europe, Brazil and Canada (Laporte et al., 2008a), where Very Small Entities (VSEs), which employed less than 25 people (Laporte et al., 2013) are the majority software companies in the respective country. Therefore in order to be always relevance in software industry, small companies need to maintains and enhances their products and for that they need to improve their development process (Valtanen & Sihvonen, 2008). Even though several methods and guidelines (e.g. Moprosoft and CMMI) have been produced in order to enhance software companies' development process, there are still a lot of challenges and obstacles have to manage (Laporte et al., 2008b). Hence, small companies whose have limited resources; particularly in financial and human resources; and practicing unique processes in managing their business have influenced their business style compare to large companies which are very formal and well documented (Sapovadia, 2006).

Therefore consider to the above characteristics and situations, have shows that most of the management processes activities (e.g. decision-making, communication and problem solving) are done in informally way (e.g. orally and less documented) and more towards to human-oriented and communication factors (Valtanen & Sihvonen, 2008; Laporte et al., 2008b). Therefore it is belief that these issues will also influence software development VSEs in organized their software development knowledge. Furthermore the influence of well organized software development knowledge is seen could assist small companies or VSEs in maintain their product relevancy in market. This process also could mitigate from knowledge atrophy problem from affecting their company.

2. RESEARCH STUDY

The study was divided into three phases: The first phase consists of a series of detailed Structured Interviews with senior management staff within the chosen organizations; whilst phase 2 entailed conducting a Focus Group with software development staff from the phase 1 companies, in order to get an understanding of the issues from a non-management perspective; Finally, phase 3 involved the distribution of a survey questionnaire to a wider set of companies than were involved in phases 1 and 2, in order to get more broad supporting data from a wider set of companies and to provide a partial validation of phase 1 and 2 findings.

The individual interview approach was used in this study in order to discuss the topics in depth, to get respondents' candid discussion on the topic and to be able to get the depth of information of the study situation for the research context (Kvale, 2007). These Structured Interviews included both open-ended and specific questions and allowed the researchers to gather not only the information anticipated, but also unexpected types of data (Li, 2006). The respondents for the individual interview session are all software development managers / CTO / owner-directors and the focus group was with software development staff. The focus group interview approach was also used in this study and aimed at collective groups of team members who are the developer of the software. An advantage of focus groups of this manner is that it allows individual team members to discuss issues in a collaborative manner with fellow team members, thus allowing a consensus to emerge which facilitates detailed data gather by researchers. Focus group interviews were also chosen because it was the most appropriate method to study attitudes and experiences; to explore how opinion was constructed (Kitzinger, 1995) and to understand behaviors, values and feelings (Patton, 2002). In order to gain more input and also to validate the above qualitative data for this study, we have developed and distributed a survey questionnaire to several Irish software VSEs. These companies were selected using personal contacts and were all directly involved in software product development, for a variety of business domains.

To ensure the participation of software development professionals who would be familiar with the considerations involved in using both software process and process improvement models, it was decided to limit the scope to software product companies whose primary business is software development. In addition, given the geographical location of the researchers, it was decided to confine the study to Irish software product companies, which has the added advantage of restricting the study to within the same economic and regulatory regime. Furthermore, restricting the study to indigenous Irish software product companies significantly increased the prospects of obtaining the historical information required to understand process foundation and evolution which would not be the case with non- Irish multinationals operating in the country, as their process would likely have been initially developed and used within the parent company prior to being devolved to the Irish subsidiary. Overall, the data collection process took 8 months, which included identifying suitable companies, contacting and confirming potential respondents' process, conducting individual and focus group interviews process and distributing and receiving questionnaires process (Basri & O'Connor, 2010).

The study data analysis process was divided into 2 main stages. In stage 1, all qualitative data gathered from individual interviews and focus groups (phases 1 and 2) was analyzed and in stage 2, the qualitative and quantitative data from the received questionnaire (phase 3) was tabulated and analyzed, with the results from this stage used to validate the analyzed results from phases 1 and 2. These 3 phases of data analysis were conducted over a four-month period.

The analysis of the qualitative data (interview and focus groups) was completed utilizing the coding mechanisms of grounded theory (Kitzinger, 1995; Elo & Kyngäs, 2008). The Grounded Theory analytical process involves a series of coding strategies, which is the process of breaking down interviews, observations and other forms of appropriate data into distinct units of meaning, which are labeled to generate concepts. These concepts are initially clustered into descriptive categories. The concepts are then re-evaluated for their interrelationships and, through a series of analytical steps, are gradually subsumed into higher-order categories, or one underlying core category, which suggests an emergent theory. Closely following the tenets of grounded theory meant that, after initial open coding, the interviews were then re-analyzed and coded axially across the higher-level categories that had emerged from earlier interviews. Any memos or propositions that emerged through the coding process were recorded

for further analysis and inclusion as questions in subsequent interviews. A consequence of this was that the interview guide was constantly updated.

3. STUDY FINDINGS

Based on the analysis process we have identified 7 main related categories that shape up the SPI environment in VSEs. Figure 2 illustrates all the categories that influence VSEs SPI initiatives. In additional these categories are the main categories and variables that gave an influence to the software development process environments in VSEs. The details of the main categories are presented below, which grouped and listed out in details the important variable that gathered from the analysis process.

3.1. Team Structure

The analysis result shows that the team environment in VSEs could be divided into 2 categories as tabulated in Figure 3. The organizational and team structure category indicates that due to small number of people working in the organization, the team size also small and this lead to a flat team and organizational structure. From the interviews analysis results indicates that all interviewees admitted that the companies having no real team structure or team structure is only exist occasionally and it depends to the company project. In additional we also found that due to the small number employee, flat organization and team structure and informal environment, interviewees are perceived that all peoples in the companies or department are in the same level. In addition the analysis show that they have the same level of working experience, skills and very much depends to each others in performing their task. Beside that the close working space or area and high frequent and informal communication are also influence this perception (Basri & O'Connor, 2011b). All these criteria have lead VSEs in narrow down the gap between the management and the team development.

Figure 2. The overall main category diagram

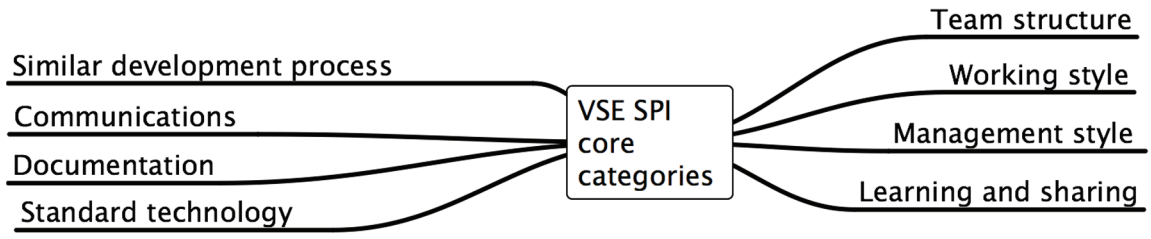
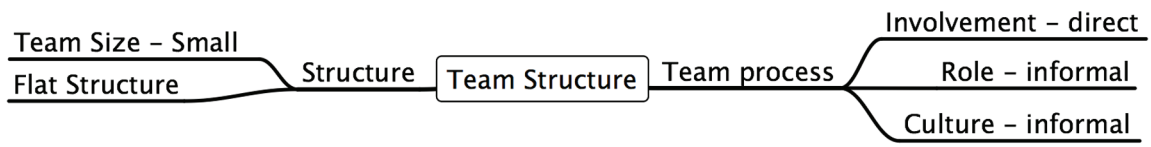


Figure 3. Team structure and process



The second category have indicates the team role, team involvement and team culture issues. The analysis shown that the staff role which includes the role in team and the task they perform in development process is very informal and very general. This could apply that the development staff could worked or assigned as different role in one time in organization development project. In addition they also can work with others or different people and different position as and when they are required. These situations have explained that team involvement process in VSEs is direct and informal in development activities.

3.2. Working and Management Style

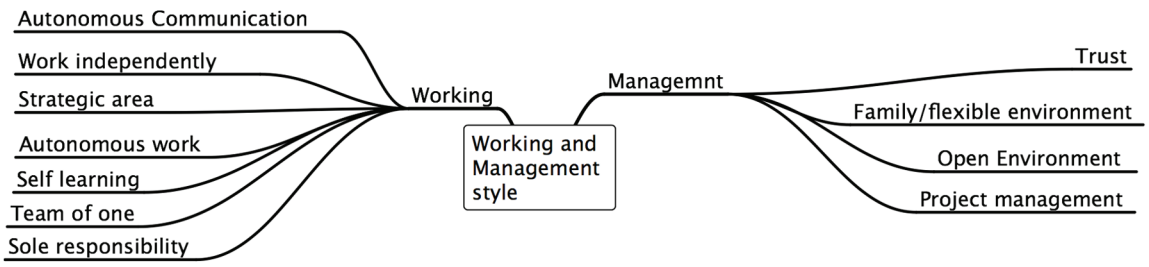
The analysis has shown that the team structure and process category gave an impact on VSEs working and management style. It is indicates that staffs have autonomy on their work which make them more self dependent, self responsibility, work independently and self learning as in Figure 4.

The result from the analysis emphasizes that people in VSEs working style is more toward individually or been assigned task according to their expertise. This situation has been defined as 'team of one' by one of the interviewee. The formal interactions of between the team member is more on the strategic area only such as problem solving or knowledge sharing in particular issue that related with the software development issues. But most of other interaction or communication are more indirect, casual and very informal. This situation gave researcher an indication that notion of team work in VSEs only appears or happened in informal way or periodic basis.

In relation to autonomous work, the analysis also indicates that, the people in VSEs also exercise an autonomous communication style in performed their works. Informal communication, less structure and direct communication, self learning and explore, frequent informal guidance, and informal meeting code that produce from the analysis indicates the autonomous communication process happened in VSEs. The analysis process also indicates that there are similar management styles adopted within VSEs. During the study, it shows that the small team size elements in VSEs are also gave an impact on the management style in the companies..

Trust, relationships, flexible environment and loose project management are the subcategories that indicate the based management style in VSEs. This type of management approaches is defined as 'Embrace and Empower' (Coleman & O' Connor, 2008b) regime as similar to 'Theory Y' management style (McGregor, 1985). In this context the idea and opinion from all subordinate have a values and been adopted in the development process and policy. There are also indicators that the element of trust in development team and their ability to carry task with less direction.

Figure 4. Working and mangement style



3.3. SPI Process

In this main category, we could be detailed into 3 categories as illustrates in Figure 5. The first subcategory that exists in process status category is process loss and focus subcategory. The results indicate the SPI process started when the process loss and/or process focus happened. Process loss happened when the technology change, customer requirement creeps, software function creep, and a new idea or suggestion from the staffs exist in their business activities. Meanwhile, process focus happened when new customer requirements, market changes, business procedure and requirement upgrade, software module or product update and expert/staff suggestion and idea occurred in their business environment. In addition, the analysis also indicates that VSEs are work very close with the customer in improving the software product and process.

The second subcategories are the small scale and informal or indirect subcategory. The analysis indicates that the SPI process in VSEs has been done in a small scale but very frequent. This process could be identify the analysis code such as organic, natural change, reactive vs. proactive, try and error, module orientation, minor changes and profitable orientate that extract from the interviews quotes represents the scale of SPI process in VSEs. Meanwhile from the analysis axial code such as RAD development, frequent change, direct and rapid change, and 'agile' process are reflected the frequent level of changes in SPI .

Beside smalls scale and high frequent changes in SPI process, the analysis also indicates that the improvement process in software development is performed in informal or indirect process. Not following any standard and guideline, not structure improvement process and informal post mortem process are the indicator that the SPI process are being performed in informal and indirect way. Meanwhile the development method category indicates that VSEs are more likely to follow an agile development approach in their software development process than the other developments method This could be identified in the communication process, documentation process, change process and customer collaboration which have been identified and explained above indicates that VSEs have fulfilled the 4 main general characteristic as in Agile Manisfesto (Fowler & Highsmith, 2001).

3.4. Communications

From the analysis, we could divide the communication process in VSEs in 2 categories namely open and informal communication category and online communication category. It also shows that the communication processes in VSEs are influence by the companies' team structure and process and the working and management style Figure 6 shows the details communication process categories produced from the analysis.

In the open and informal category, we have identified several interviews quotations that indicate the communication process where people are more towards informal and direct/casual communication. This

Figure 5. SPI process

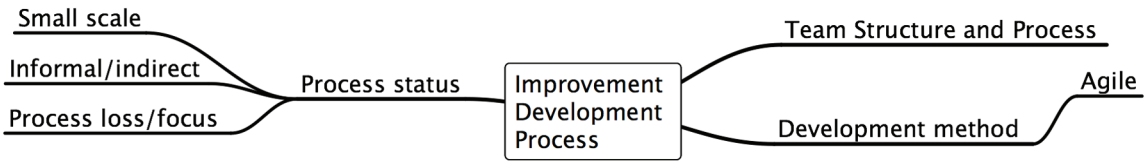
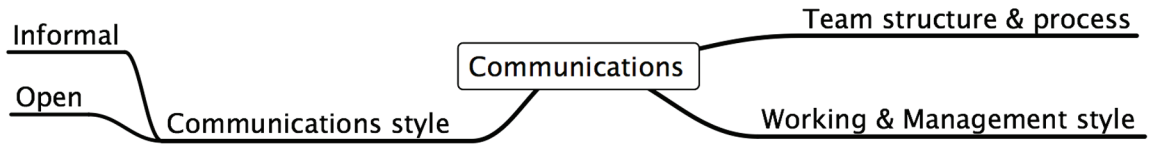


Figure 6. Communication process



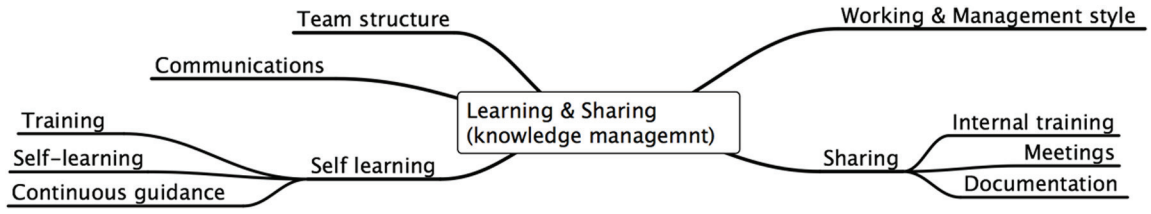
could be identifies in the way of meeting have been conducted which are more informal, 'stand up', periodic and individual. This is due to the working environment, team size and working style in their company. Furthermore relationship between staffs in the company also influences the communication process in VSE. The family and flexible environment, frequent socialize between staff; flat organization structure and closeness working space have given an impact on communication process in VSEs. The analysis also shows that the use of communication tools such as email, phone, blog and internet are very active in VSEs. This communication tools is more vital to the company that have a staff who works outside Ireland or having others offices in different locations. The use of these tools is believed could close the gap between remote and collocate staff and allow staffs to share and document all work related information or knowledge in informal way.

3.5. Learning and Sharing Process

The analysis has shows the learning and sharing process in VSEs as in Figure 7. In self learning category, the analysis shows at in VSEs there are no formal trainings are given or provides to employees in enhancing their knowledge or skills. The analysis also has explained that people in VSEs are more depends to self learning in mastering the technology or process that used in the organization. Besides self learning, the analysis also shows on the job training, self exploring and continues guidance from expert with in the companies are the main process that frequently been practiced in enhanced staff knowledge and skills.

The second category in this part is sharing category. The analysis shows that in VSEs knowledge sharing process happened in informal training, informal meeting and document sharing. Informal training happened through informal and guidance from expert, peer to peer programming process, shared books and others material, internal training, high frequent open and direct discussion with team member and online sharing with others. Meanwhile the informal meeting process happened through an informal stand-up meeting, direct and open discussion and online meeting via email, skype and blog. In relation, the analysis results indicate that the learning and sharing process in VSEs is been influenced and shaped by 3 existing main factors which are VSEs team size and process which are small team size and flat orga-

Figure 7. Learning and sharing process



nization structure; working and management style which are more toward autonomous work and macro management process and, communication process which are indirect and informal process. In additional from the interviews data analysis shows that in general knowledge sharing activities either via electronic or personal are important in maintaining and evolving the current VSEs software development process.

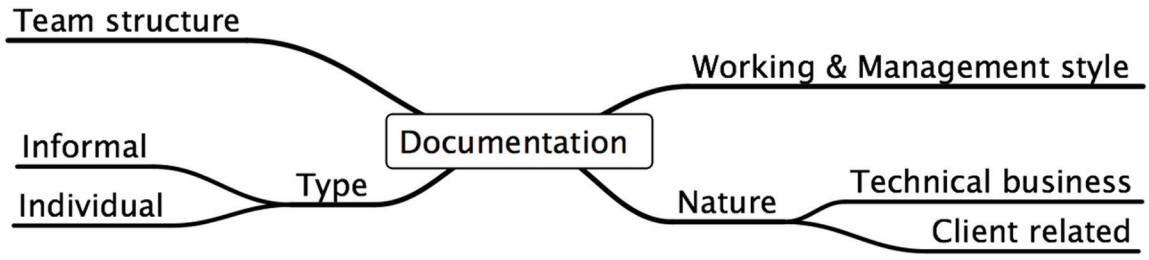
3.6. Documentation Process

The results have indicates 2 category that falls in main documentation activities as in Figure 8. The analysis has indicates the documentation processes are very informal process and individual initiatives. In additional, in VSEs documentation process are not given high priority because of time constraints and small team size. The results from the analysis also indicate in VSEs most of the information is documented in an electronic format rather than a paper format. The interviewees also admit due to similar technology and development method applied in all development projects, details documentation process is not necessary and important. They claimed that the staffs are more focused to software development activities rather than the documentation process. In additional from the analysis indicates that due to the autonomous work culture in VSEs, which based on person experience and skills, most of the documentation process in VSEs is individual and personal basis. Beside that, the analysis also shows that programming codes, technical issues and business procedures are the main documentation in the VSEs which fall under specific information and procedure category.

4. DISCUSSION

From the overall discussion above, a theoretical and relational model has been presented which shows that the software process and process improvement strategy which started from process loss or process focus which was influences by several variables which has been discuss above. The process formation is created and will indicate the process that need to improve, change or upgrade. As discussed above the software development process in VSEs are done in informal, indirect and small scale at one time but in a high frequent. The analysis in this part also shows that VSEs followed or adopted the agile development approaches, which involved a lot of interaction or communication either with the customers or the developers, high focus of the development process and having minimal documentation process the organization. Due to small team size, flat team and organization structure, staffs or management geographical location, autonomous working style and macro project management the communication

Figure 8. Documentation process



process are more become informal and autonomous. Beside that the uses of the communication tools are also being used extensively among peoples in communicate and shared their knowledge. In term of documentation process, the analysis indicate that in VSEs the documentation process have been practiced either informal or individual. The analysis also shows that due to macro project management style, autonomous working style and influence standard technology and similar development process have lead to the these situation. Moreover the interviewees admitted that only the issue related to business procedure and technical specification are being formally documented. Moreover the analysis also indicated that due to the informal communication, informal documentation and autonomous work have created the informal and personal organization learning and sharing process. Therefore from learning and sharing process a new idea and weakness of the area that need to improved, change and upgrade. This process will start back at the process formation and iterative.

4.1. Contribution

It was collectively agreed by the respondents that the documentation process in VSEs is done very informally, individually and specifically. In term of knowledge management issues, the result showed that all respondents claimed that they have a clear KM strategy in the organization. However the analysis showed that this process are done informally and is not organised. In addition the result show that even though the KM was done informally either in communication, management, working style and team structure in VSEs, 90% of the respondents believed that this environment have lead them to mitigate the knowledge and process loss problem in their organization. Moreover the results also indicated that in overall the size of the company given an impact to all the process that have discussed above.

The main contribution of this study is an expanded understanding of SPI research area by merging the issues of KM from both a general and VSE specific perspective. Our results indicate that KM factor gave indirect influences to the process of improving current software process and process improvement activities in software development companies. Other contributions of this research is in providing an additional knowledge to the SPI research area focused more on VSEs, which have been least explored by current literature (O'Connor, 2012). The research has found the variables that influence the software process and process improvement issues that could be explored individually in further detail in future. The last contribution is the type of strategies used to carry out research methodology work, especial in analyzing the qualitative data, which was the output from the interviews (interview and focus groups) activities. If survey questionnaire approach is a familiar approach and was often used in the software engineering field, the interviews data analysis research technique, which adopted qualitative contents data analysis and GT coding approach, is rarely been used in the analysis of the software process improvement and in software engineering research in general. Therefore we believe that we are adding to the body of knowledge associated with suitability of the GT research method to software engineering area.

4.2. Future Work

There are a number of potential avenues of further research related to this study. Of primary interest to the researchers is to widen the current research spectrum. Specifically, to test current research findings and also to produce and provide more valid findings and results, a similar study could be deployed in other geographical locations. This could help to create more generalizable research findings and assist

with validation of the present research. In addition, the involvement of non-IT companies having a small IT department could assist future researchers to compare and produce a pattern of research results which could also add to the present research.

ACKNOWLEDGMENT

This work was supported, in part, by Science Foundation Ireland grant 10/CE/I1855 to Lero - the Irish Software Engineering Research Centre (www.lero.ie).

REFERENCES

Aaen, I., Arent, J., Mathiassen, L., & Ngwenyama, O. (2002). A conceptual map of software process improvement. *Scandinavian Journal of Information Systems*, *13*(13), 81–101.

Aaen, I., Börjesson, A., & Mathiassen, L. (2007). SPI agility: How to navigate improvement projects. *Software Process Improvement and Practice*, *12*(3), 267. doi:10.1002/spip.309

Aurum, A., Jeffery, R., Wohlin, C., & Handzic, M. (2003). *Managing software engineering knowledge*. Germany: Springer. doi:10.1007/978-3-662-05129-0

Awazu, Y. (2004). Knowledge management in distributed environments: Roles of informal network players. In *Proceedings of the 37th Annual Hawaii International Conference on System Sciences*, (p. 6).

Basri, S., & O' Connor, R. V. (2010). Evaluation of knowledge management process in very small software companies: A survey. In *Proceedings of the 5th International Conference on Knowledge Management,* Kuala Terengganu, Terengganu, Malaysia (pp. 1-6).

Basri, S., & O'Connor, R. (2011a). Knowledge management in software process improvement: A case study of very small entities. In M. Ramachandran (Ed.), *Knowledge engineering for software development life cycles: Support technologies and applications* (pp. 273–288). IGI Global. doi:10.4018/978-1-60960-509-4.ch015

Basri, S., & O'Connor, R. V. (2011b). A study of software development team dynamics in SPI. In R. O'Connor, J. Pries-Heje, & R. Messnarz (Eds.), *Systems, software and services process improvement (CCIS)* (Vol. 172, pp. 143–154). Springer-Verlag. doi:10.1007/978-3-642-22206-1_13

Bjornson, F. O., & Dingsoyr, T. (2005). *A study of a mentoring program for knowledge transfer in a small software consultancy company (LNCS 3547)*. Berlin / Heidelberg, Germany: Springer.

Bjørnson, F. O., & Dingsøyr, T. (2008). Knowledge management in software engineering: A systematic review of studied concepts, findings and research methods used. *Information and Software Technology*, *50*(11), 1055–1068. doi:10.1016/j.infsof.2008.03.006

Borges, L. M. S., & Falbo, R. A. (2002). Managing software process knowledge. In *Proceedings of the International Conference on Computer Science, Software Engineering, Information Technology, e-Business, and Applications (CSITeA'2002)* (pp. 227–232).

Clarke, P., & O'Connor, R. V. (2012). The situational factors that affect the software development process: Towards a comprehensive reference framework. *Journal of Information and Software Technology*, *54*(5), 433–447. doi:10.1016/j.infsof.2011.12.003

Coleman, G., & O'Connor, R. V. (2008a). Investigating software process in practice: A grounded theory perspective. *Journal of Systems and Software*, *81*(5), 772–784. doi:10.1016/j.jss.2007.07.027

Coleman, G., & O'Connor, R. V. (2008b). The influence of managerial experience and style on software development process. *International Journal of Technology*. *Policy and Management*, *8*(1), 91–109.

Dingsoyr, T., & Conradi, R. (2002). A survey of case studies of the use of knowledge management in software engineering. *International Journal of Software Engineering and Knowledge Engineering*, *12*(4), 391–414. doi:10.1142/S0218194002000962

Dreyfus, H. L., Dreyfus, S. E., & Athanasiou, T. (1986). *Mind over machine: The power of human intuition and expertise in the era of the computer*. New York, NY: Free Press.

Edwards, J. S. (2003). Managing software engineers and their knowledge. In A. Aurum, R. Jeffery, C. Wohlin, & M. Handzic (Eds.), *Managing software engineering knowledge* (pp. 5–27). New York, NY: Springer. doi:10.1007/978-3-662-05129-0_1

Elo, S., & Kyngäs, H. (2008). The qualitative content analysis process. *Journal of Advanced Nursing*, *62*(1), 107–115. doi:10.1111/j.1365-2648.2007.04569.x PMID:18352969

Fowler, M., & Highsmith, J. (2001). The agile manifesto. *Software Development*, *9*(8), 28–35.

Grant, R. M. (1996). Prospering in dynamically-competitive environments: Organisational capability as knowledge integration. *Organization Science*, *7*(4), 375–387. doi:10.1287/orsc.7.4.375

Hansen, B. H., & Kautz, K. (2004). *Knowledge mapping: A technique for identifying knowledge flows in software organisations (LNCS 3281)*. Springer. doi:10.1007/978-3-540-30181-3_12

Hendricks, P. H. J., & Vriens, D. J. (1999). Knowledge-based systems and knowledge management: Friends or foe. *Information and Management Journal*, *35*, 113–125. doi:10.1016/S0378-7206(98)00080-9

Humphrey, W. S. (1989). *Managing software process*. Reading, MA: Addison Wesley.

Jeners, S., Clarke, P., O'Connor, R. V., Buglione, L., & Lepmets, M. (2013). Harmonizing software development processes with software development settings – A systematic approach. In F. McCaffery, R. V. O'Connor, & R. Messnarz (Eds.), *Systems, software and services process improvement (CCIS 364)*. Springer-Verlag. doi:10.1007/978-3-642-39179-8_15

Kettunen, P. (2003). Managing embedded software project team knowledge. *IEEE Software*, *I*(6), 359–366. doi:10.1049/ip-sen:20031173

Kitzinger, J. (1995). Introducing focus groups. *British Medical Journal*, *311*, 299–302. doi:10.1136/bmj.311.7000.299 PMID:7633241

Kruchten, P. (2000). *The rational unified process*. Reading, MA: Addison Wesley.

Kukko, M., Helander, N., & Virtanen, P. (2008). Knowledge management in renewing software development processes. In *Proceedings of the 41st Annual Hawaii International Conference on System Sciences* (pp. 332-332).

Kvale, S. (2007). *Doing interviews: The Sage qualitative research kit*. Thousand Oaks, CA: Sage.

Laporte, C. Y., Alexandre, S., & O'Connor, R. (2008). A software engineering lifecycle standard for very small enterprises. In *Proceedings of the 15th European Conference on Software Process Improvement* (Vol. 16, pp. 129-141). Berlin, Germany: Springer-Verlag.

Laporte, C. Y., Alxender, S., & Renault, A. (2008). Developing international standards for very small enterprises. *Journal of Computer*, *41*(3), 98. doi:10.1109/MC.2008.86

Laporte, C. Y., O'Connor, R. V., & Fanumy, G. (2013). International systems and software engineering standards for very small entities. *CrossTalk. The Journal of Defense Software Engineering*, *26*(3), 28–33.

Li, J. Y. (2006). *Process improvement and risk management in off-the shelf componenet-based development*. Unpublished doctoral dissertation, Norwegian University science and Technology.

Litern, G., Diedrich, F. J., & Serfaty, D. (2002). Engineering the community of practice for maintenance of organizational knowledge. In *Proceedings IEEE 7th Conference on Human Factors and Power Plants* (pp. 7-13).

Mathiassen, L., & Pourkomeylian, P. (2003). Managing knowledge in a software organization. *Journal of Knowledge Management*, *7*(2), 63–80. doi:10.1108/13673270310477298

McGregor, D. (1985). *The human side of enterprise: 25th anniversary printing*. McGraw-Hill/Irwin. doi:10.1016/B978-0-12-054752-4.50008-7

Nonaka, I., & Takeuchi, H. (1995). *The knowledge creating company*. New York, NY: Oxford University Press.

Nonaka, I., Toyama, R., & Konno, N. (2000). SECI, Ba and leadership: A unified model of dynamic knowledge creation. *Long Range Planning*, *33*(1), 5–34. doi:10.1016/S0024-6301(99)00115-6

O'Connor, R., & Coleman, G. (2009). Ignoring 'best practice': Why Irish software SMEs are rejecting CMMI and ISO 9000. *Australasian Journal of Information Systems*, *16*(1), 7–30.

O'Connor, R. V. (2012). Evaluating management sentiment towards ISO/IEC 29110 in very small software development companies. In Mas, A., Mesquida, A., Rout, T., O'Connor, R. V., & Dorling, A. (Eds.), Software process improvement and capability determination, (CCIS Vol. 290). Springer-Verlag.

Parent, M., Gallupe, R. B., Salisbury, W. D., & Handelman, J. M. (2000). Knowledge creation in focus group: Can group technologies help? *Information & Management*, *38*(1), 47–58. doi:10.1016/S0378-7206(00)00053-7

Patton, M. Q. (2002). *Qualitative evaluation and research methods* (3rd ed.). Newbury Park, CA: Sage Publications, Inc.

Rhodes, J., Hung, R., Lok, P., Lien, B. Y., & Wu, C. M. (2008). Factors influencing organizational knowledge transfer: Implication for corporate performance. *Journal of Knowledge Management*, *12*(3), 84. doi:10.1108/13673270810875886

Ryan, S., & O'Connor, R. (2009). Development of a team measure for tacit knowledge in software development teams. *Journal of Systems and Software*, *82*, 229–240. doi:10.1016/j.jss.2008.05.037

Ryan, S., & O'Connor, R. (2013). Acquiring and sharing tacit knowledge in software development teams: An empirical study. *Information and Software Technology*, *55*(9), 1614–1624. doi:10.1016/j.infsof.2013.02.013

Sapovadia, V. K. (2006). *Micro finance: The pillars of a tool to socio-economic development.* Development Gateway. Retrieved on October 9, 2008, from http://ssrn.com/abstract=955062

Shaw, D., Edward, J. S., Baker, B., & Collier, P. M. (2003). Achieving closure through knowledge management. *Electronic Journal of Knowledge Management*, *1*(2), 197–204.

Sirvio, K. S., Mantyniemi, A., & Seppanen, V. (2002). Toward a practical solution for capturing knowledge for software projects. *IEEE Software*, *19*(3), 60–62. doi:10.1109/MS.2002.1003457

Szulanski, G. (1996). Exploring internal stickiness: Impediments to the transfer of best practice within firm. *Journal of Strategic Management*, *17*(2), 27–43.

Turner, K., & Makhija, M. (2006). The role of organizational controls in managing knowledge. *Academy of Management Review*, *31*(1), 197–217. doi:10.5465/AMR.2006.19379631

Valtanen, A., & Sihvonen, H. M. (2008). Employees' motivation for SPI: Case study in a small Finnish software company. In *Proceeding of the 15th European Conference (EuroSPI 2008)* (pp. 152-163). Berlin/Heildelberg, Germany: Springer–Verlag.

Weick, K. E. (1995). *Sense making in organization*. Thousand Oak, CA: Sage Publication.

Zahran, S. (1998). *Software process improvement: Practical guidelines for business success*. Boston, MA: Addison Wesley.

This research was previously published in the International Journal of Systems and Service-Oriented Engineering (IJSSOE), 4(1); edited by Dickson K.W. Chiu, pages 39-52, copyright year 2014 by IGI Publishing (an imprint of IGI Global).