Requirements Engineering in Virtual Software Development: Achieving Balance

Jo Hanisch

School of Accounting and Information Systems, University of South Australia, Adelaide, South Australia 5000
Tel: 61 8 8302 0252, Fax: 61 8 8302 0992, Email: jo.hanisch@unisa.edu.au

ABSTRACT

There has been growing interest in virtual teams, and more specifically in virtual software development. Requirements engineering, which is seen as a crucial phase in software development provides another dimension when software development occurs in a virtual setting. While formal software development methods are the obvious first choice for project managers to ensure a virtual information system project team remains on track, the social aspects of requirements engineering cannot be ignored. These social aspects are especially important across different cultures, and have been shown to affect the success of an information system. This paper proposes a framework indicating that project managers need to encourage a balance between formal methods and social aspects in requirements engineering to suit the virtual team members.

INTRODUCTION

Within contemporary organisations it is usual to find an array of computer-based information systems which support their business processes and assist the organisations to achieve their business goals. Businesses have changed as predicted by Drucker in 1988. Organisations have a flatter structure with "decentralized and autonomous units" including the extensive use of "task forces" (Drucker 1988). Given the new organisational structure in business, and the more recent emergence of the global economy and global markets (Karolak 1998), there is a change in both the type of work being performed and the way work is developed. For example, many companies operate entirely using the Internet capabilities (Laudon and Laudon 2000).

As businesses have changed, so have their information systems. Information systems (IS) still support organisations as they achieve their business goals, however both the type of systems being developed and the nature of software development have changed. There is evidence of change from locally developed software to virtual (or global) software development (Carmel 1999). As the technology now exists to enable collaborative team work over distance and time, virtual teams, in which virtual software development occurs, are becoming more recognised as the rule rather than the exception (Kimball 1997). There is no doubt that virtual software development teams present some challenges to organisations. However, with improved understanding of the influences during virtual software development, organisations have the opportunity to develop strategies and techniques to manage this type of development. This in turn can help reduce the risk of failure of the systems being developed.

As Kuiper (1998) states, one of the greatest challenges for business is defining its needs for a new information system. Kuiper (1998) continues that the complexity and factors to be considered in developing a new information system is often overwhelming to business. Added to this is the realisation that the success or failure of the new information system, often determines whether the business succeeds or fails. Therefore ensuring the needs of business are clearly determined and communicated is important in the process of developing an information system.

One particularly crucial phase in software development that requires clear communication is the requirements definition (engineering) phase (Darke and Shanks 1997). This phase has been said to impact directly on the success or failure of new IS in organisations (Byrd et al 1982; Davis 1990). Requirements engineering occurs early in the software development process, where the requirements for an information system are defined and expressed in the form of a systems requirements specification (Greenspan et al 1994).

The main purpose of this paper is to explore some of the influences associated with the requirements engineering phase of the software development process as it occurs in a virtual setting. The paper first reviews the concept of virtual teams and virtual software development, then the requirements engineering phase of software development is reviewed, followed by a discussion of requirements engineering in virtual software development. This paper is offered as a starting point for debate concerning two opposing forces perceived in the IS literature within requirements engineering and virtual software development

VIRTUAL TEAMS AND VIRTUAL SOFTWARE DEVELOPMENT

Jarvenpaa and Ives (1994) describe successful organisations as those that are moving towards a dynamic network structure. They suggest that "dynamic network organisations are spun from small, globally dispersed, ad hoc teams or independent organisational entities performing knowledge or service activities"(Jarvenpaa and Ives 1994:26). In this way organisations can respond quickly and efficiently to customer demands or changes in the external environment. Hacker and Kleiner (1996) support the notion that the extent to which an organisation is dynamic in an agile, competitive environment will affect its success.

Many organisations are already planning to operationalise the necessary human and information technology resources to cultivate the growth of the virtual environment (Greenbaum, 1998). Grenier and Metes (1995) state that some of the factors driving the trend towards virtual teams include mergers, corporate acquisitions, downsizing and increased use of new technology. As a result of downsizing on a global scale by the organisation, much of their expertise becomes spread throughout the organisation across a number of countries or regions, so virtual teams may form. Significantly many of the mergers and acquisitions involve international offices, and this also spawns the development of virtual teams with members potentially from many different cultures. When business requires certain projects to be undertaken in the organisation, virtual teams are established utilising expertise from any area of the organisation without the necessity for re-location of staff (Melymuka 1997, Hacker and Kleiner 1996).

Hacker and Kleiner (1996:196) believe that virtual teams have progressed from the status of a "good idea" to "a critical

strategy for many organisations". From these authors, it appears that virtual teams are now being considered as a natural option to address business problems when team members are geographically separated. Melymuka (1997) supports the idea that virtual teams have been touted as one way to circumvent problems associated with re-location of team members to a central business site. This has been particularly evident with software development as there has been a dearth of skilled IS specialists globally over the past five years (Carmel 1999).

What are virtual teams and why are they important to management? "Virtual teams are groups of people working closely together even though they may be separated by many miles, even continents" (Beger 1995:36). Hartman and Guss (1996:185) more fully define virtual teams as *"a temporary network of independent professionals, separated by geographic, temporal and psychological distance, whose use of telecommunications tools for business communication is interdependent, to satisfy the business requirement of sharing skills and working to meet a common goal"*.

Virtual teams are characterised by members who are physically isolated, who interact mainly through the use of computer-mediated communication technologies and who rarely or never meet face-to-face (Jarvenpaa and Leidner 1997). Virtual teams are said to be functional units of an organisation which are flexible, and which both quickly and professionally "execute multiple projects, anywhere and anytime" (Hartman and Guss 1996:185). They are characterised by short-term (i.e., 6 months) project based work and by their nature virtual teams lend themselves to ad hoc teams or "task-force" work groups which come together for specific project objectives and disband once these have been fulfilled (Jarvenpaa and Leidner 1997).

There is recognition of barriers to virtual teams, such as, information overload and its associated stress, decreased informal communication capabilities, resistance to new work methods, uncertainty with requirements, and uncertainty with security, control, power and information when working with external organisations (Grenier and Metes 1995).

While there are barriers to virtual team work, there are advantages such as the decrease in time and cost of travel, the ability to select expertise from the whole organisation regardless of employees' locations and the speed and flexibility at which project teams can be formed and disbanded (Lipnack and Stamps 1997).

Virtual (or global) software development teams are but one type of an array of possible virtual teams. Knowledge in the area of virtual software development is increasing, however this is recognised as an emerging field of research (Carmel, 1999). Issues which appear most frequently in the limited literature concerning virtual software development include loss of communication richness (Jarvenpaa and Leidner 1997; Carmel 1999), cultural differences (Carmel 1999), loss of identity with the team (Karolak 1998; Carmel 1999) and lack of management support (Karolak 1998).

There are more questions and issues than answers at this stage, and many more questions will no doubt present themselves over the next decade, particularly when implementation and maintenance of the software newly developed by virtual teams stands the test of time. At this point many organisations are still questioning whether this type of development is suitable for them, or whether they should wait to determine the success of the software developed by virtual teams.

Studies concerning the management of virtual software development (Carmel 1999; Karolak 1998) indicate that formalising the software development methods is essential in virtual software development. Other researchers (Jones 1994; Meadows 1996; Grenier and Metes 1995) concur that virtual work requires a more formalised regime of work plans, deliverable progress reports and mandatory meeting schedules to assist the team to remain on track and focused for the delivery of the product. Researchers in the area of computer-based communication media (Damian et al 2000; Prabhakar 1999; Jarvenpaa and Leidner 1997) also consider that formalised procedures aid in the success of communication between virtual team members and in the ultimate success of the project.

Given the research concerning the management of virtual teams and that of computer-based communication technologies, it appears that formal software development methods are not only essential, but of utmost priority, in ensuring the successful delivery of a new information system in an organisation. However, before stating the proposition that formal techniques are the ultimate project management tools for virtual software development and that project managers should be rigorous in their applications of formal methods, the influencing factors during requirements engineering (as part of software development) need to be considered.

REQUIREMENTS ENGINEERING

Requirements engineering has been defined as "the disciplined application of scientific principles and techniques for developing, communicating, and managing requirements" (Christel and Kang 1992:3). This definition of requirements engineering is supported by the well known definition from Loucopoulos and Karakostas (1995), which is defined as:

"...the systematic process of developing requirements through an iterative co-operative process of analysing the problem, documenting the resulting observations in a variety of representation formats, and checking the accuracy of the understanding gained." Loucopoulos and Karakostas (1995:13)

Requirements engineering therefore covers "all the activities of discovering, documenting, and maintaining a set of requirements for building a computer-based information system" (Thanasankit and Corbitt 1999; Sommerville and Sawyer 1997). During requirements engineering, the requirements engineer (or systems analyst) needs to be mindful of the objectives/outcomes of the requirements engineering phase and implement appropriate processes or techniques which will help to avoid failure of the IS (Macauley 1996). But the failure of many of these IS development projects is due not just to the inadequate requirements (Boehm 1981) in general, but more specifically to the social, political and cultural factors associated with the project (Goguen and Linde 1993).

The development of an "effective IS requires thorough analyses of user information needs prior to IS design" (Byrd et al 1992:117). Requirements engineering, which is concerned with understanding the needs of the client (user) and determining the systems requirements which satisfy these needs, given any identified constraints and exclusions (Carroll and Swatman 1997), is a crucial phase in the software development process (Greenspan et al 1994).

So why is the requirement engineering phase so important to IS specialists and users? As Carroll and Swatman (1997:2) state, any "inconsistencies, omissions and errors in the requirements specification have significant impact upon the ability of the developed systems to meet customers' needs." The implication is that the client is less likely to accept and use the system that has been developed (Hocking 1996) when there are problems with either the communication or agreement of the requirements (Urquhart 1997). In project management terms of delivery of the system on time, within budget and meeting the users' requirements (McLeod and Smith 1996), the system is said to have failed to deliver. Laudon

and Laudon (2000) attribute this to failure of the organisation to meet its goals. Further, most errors found later in the systems development process can be retraced to the requirements specification, which either contains errors in the requirements or misinterpretation of the requirements (Liou and Chen 1994).

The significance is that early detection of errors or misinterpretations decreases the cost of their rectification later in the process (Boehm 1981; Park, Kim and Ko 2000) and improves the quality of the system being developed (Darke and Shanks 1997a). As Thanasankit (1999) has shown, these costs can be substantial to an organisation. According to Park et al (2000:429), 5% of the total cost of a major system is spent on its design and development, while "70% of the ability to influence the quality comes from this meager amount". They state that errors in the requirements which are left uncorrected early in the software development cycle, ultimately end up as maintenance problems in the delivered software. It transpires that 5% of the total effort expended on requirements analysis leverages 50% of the total influence on improved quality of the system (Park et al 2000).

Within the social context, it is necessary for IS specialists to understand communication and cooperation, as well as social complexity during requirements engineering (Thanasankit 1999). Requirements engineering research has traditionally been positivist in its approach, largely focusing on the methods and tools used in the gathering elicitation and validation of requirements. Many researchers conclude that the more formal the techniques used, the more likely that the requirements will be clearly defined and understood. However, as Thanasankit (1999) states, organisations need to consider the emotions and culture of users and IS specialists. It has been shown that different cultures will perform tasks not only because they are responsible for the task, but because they wish to maintain surface harmony and trust between the group (Thanasankit 1999) and this often inhibits formal sign offs of requirements specifications, which in turn causes delays and potential failure of the project. In these cases, the imposition of formal, often western philosophies and methods, have a negative impact on the requirements gathering process (Thanasankit 1999).

REQUIREMENTS ENGINEERING IN VIRTUAL SOFTWARE DEVELOPMENT

There may occur a dilemma for project managers who are responsible for the requirements engineering phase during virtual software development. When systems analysts use more formalised tools and techniques for gathering requirements, as these tools are deemed to assist in the process of virtual communications, there may occur a negative impact on the social process of gathering of the requirements.

There may arise tension between the two forces and therein lies the interest and motivation for this discussion. Virtual work and virtual teams have been shown by previous research (Carmel 1999; Karolak 1998; Jones 1994; Meadows 1996; Grenier and Metes 1995) to benefit from a more structured approach to the working environment. Therefore it would be logical to deduce that the requirements engineering phase of systems development in a virtual environment would also benefit from a more structured approach.

However, western ideologies created requirements engineering foundations (Odedra 1993) and many of these processes were formalised and structured to suit western philosophies. Jirotka and Goguen (1994) argue that requirements engineering should be developed in an environment which suits both the social and technical concerns of the requirements. These structured methods may not be suitable for members of a virtual team who are likely to be situated in regions and cultures where social processes are deemed more significant in the requirements gathering process, rather than utilising structured tools and techniques (Thanasankit 1999).

From the reviews above, many virtual teams include members who are from different cultural and social backgrounds. They will come into the team with different experiences in the formal techniques of software development and the way they have been used in their region previously. This will affect the way they communicate the requirements of the system and the extent to which formal method are used. Requirements engineering in western cultures will be different from eastern cultures.

Figure 1 below shows a framework representing the tension which may occur between the two forces of social aspects of requirements engineering and formal software development tools used in requirements engineering.

Figure 1: A Framework of Requirements Engineering in Virtual Software Development (adapted from McLeod and Smith 1996:6)
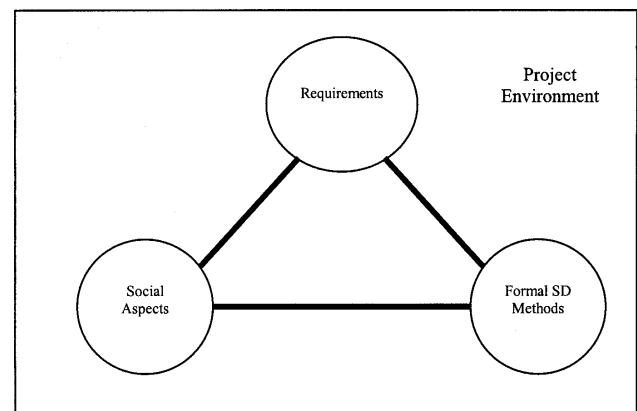
The diagram indicates that the project manager who controls the project environment will always be balancing these variables to produce the requirements for the information system being developed. Therefore it is impossible to move either of the sides of the triangle without impacting on the requirements. For example, if highly formal software development methods are chosen at the expense of social considerations the requirements will be affected, and vice versa. The onus will be on the project manager of the virtual team to create a balance between social processes and structured methods in order to gather adequately the needs of the users and keep the project on track.

CONCLUSION

Requirements engineering is a crucial stage in software development regardless of whether the development is local or global. During virtual software development previous research has recommended that more formal and structured methods of requirements gathering be employed to decrease the risk of misunderstanding or misinterpretation of the requirements. However, requirements engineering is a social process and formal, structured methods may impact on the social and cultural differences between virtual teams members.

Project managers therefore face the task of creating a project environment that is conducive to communication through electronic means and which takes into account the differences in cultural and social aspects of virtual team members as they communicate their requirements. A balance is required to satisfy the formal struc-

tured methods of requirements engineering which have been shown in the past to improve software development in project management terms of quality, cost and time, and the need to provide a project environment which encourages open communication and supports cultural differences in virtual software development. Then not only will the system be built right but the right system will be built. Further research is required to determine which requirements engineering processes and which project environment characteristics are more appropriate in helping to achieve this balance.

REFERENCES

Beger G (1995) Virtual Teams, *Training*, April, p36.

Boehm B (1981) *Software Engineering Economics*, Prentice-Hall, New Jersey.

Byrd T A, Cossick K L and Zmud R W (1992) A synthesis of research on requirements analysis and knowledge acquisition techniques *MIS Quarterly* March, pp 117-138.

Carmel E (1999) *Global software teams: collaborating across borders and time zones*, Prentice Hall: NJ

Carroll J M and Swatman P A (1997) 'How can the requirements engineering process be improved?', in Sutton D J (ed): *Proceedings of the 8th Australasian Conference on Information Systems*, University of South Australia, Adelaide, South Australia.

Christel M G and Kang K C (1992) Issues in requirements elicitation, Technical Report CMU/SEI-92-TR-12, SEI Pennsylvania.

Damian D E H, Shaw M L G and Gaines B R (2000) 'A study in requirements negotiations in virtual project teams', *Proceedings of the 8th European Conference on Information Systems 2000*, Vienna Austria.

Darke P and Shanks G (1997) Managing user viewpoints in requirements definition, *Proceedings of the 8th Australasian Conference on Information Systems*, Adelaide, Australia.

Davis A M (1990) *Software requirements analysis and specification*, Prentice Hall: NJ

Drucker P (1988) The coming of the new organisation, *Harvard Business Review*, Jan-Feb, 47-53

Goguen J A and Linde C (1993) 'Techniques for requirements elicitation', *Proceedings of the IEEE International Symposium on Requirements Engineering*, IEEE Computer Society Press, USA, pp. 152-164.

Greenbaum T L (1998) 'Avoiding a virtual disaster' *HR Focus* 75(2): 11-12

Greenspan S, Mylopoulos J and Borgida A (1994) 'On formal requirements modeling languages: RML revisited, In *Proceedings 16th International Conference on Software Engineering*, Sorrento, May

Grenier R and Metes G (1995) *Going Virtual: Moving your Organisation into the 21st Century*, NJ: Prentice Hall.

Hacker M E and Kleiner B M (1996) Identifying Critical Success Factors Impacting Virtual Group Performance, *Proceedings of IEMC 96. Managing Virtual Enterprises: A Convergence of Communications, Computing and Energy Technologies*. 18-20 August 1996, Vancouver, BC, Canada. IEEE publication, 196-200.

Hartman F and Guss C (1996) Virtual Teams - Constrained by Technology or Culture, *Proceedings of IEMC 96. Managing Virtual Enterprises: A Convergence of Communications, Computing and Energy Technologies*. 18-20 August 1996, Vancouver, BC, Canada. IEEE publication, 645-650.

Hocking L (1996) 'Systems analysis and early design as the negotiation of meaning and interest' *Proceedings of the 7th Australasian Conference on Information Systems*, Tasmania, Australia.

Jarvenpaa S L and Ives B (1994) The Global network Organisation of the Future: Information Management Opportunities and Challenges, *Journal of Management Information System,* 10(4), 25-57.

Jarvenpaa S L and Leidner D E (1998) Communication and Trust in Global Virtual Teams, *Journal of Computer-Mediated Communication*, 3 (4), http://www.ascusc.org/jcmc/vol3/issue4/jarvenpaa.html.

Jirotka M and Goguen J (1994) *Requirements Engineering – Social and Technical Issues*, Academic Press, London, UK.

Jones C (1994) Globalisation of software supply and demand, *IEEE Software*, Nov, 11(6):17-24

Kimball L (1997) *Managing virtual teams*, Teams Strategies Conference, Toronto Canada URL: http://www.tmn.com/~lisa/vteams-toronto.htm [on-line accessed 20 Sep 2000]

Kuiper D (1998) 'The art of defining computer system requirements: say what you need and need what you say' *Hospital Materiel Management Quarterly*, May pp 14-21.

Laudon K C and Laudon J P (2000) *Management information systems: organisation and technology in the networked enterprise*, 6th edition, Prentice Hall:NJ

Liou Y I and M Chen (1994) 'Using group support systems and joint application development for requirements specification' *Journal of Management Information Systems* 10(3): 25-41.

Lipnack J and Stamps J (1997) *Virtual Teams: Reaching Across Space, Time, and Organizations with Technology*. New York: John Wiley & Sons, Inc.

Loucopoulos P and Karakostas V (1995) *System Requirements Engineering*, McGraw-Hill Book Company Europe, Berkshire, UK.

Macaulay L A (1996) *Requirements Engineering*, Springer-Verlag, London.

McLeod G and Smith D (1996) *Managing information technology projects*, Boyd & fraser, ITP Publishing Company: USA

Meadows C J (1996) 'Globalising software development' *Journal of Global Information Management* 4(1):5-15

Melymuka K (1997) Virtual Realities, *Computerworld*, 28 April, 71-73.

Odedra M (1993) 'Enforcement of foreign technology on Africa: its effect on society, culture and utilisation of information technology', in Beardon C and Whitehouse D (eds) (1993) *Computers and Society*, Intellect, UK

Park S H, Kim and (2000) 'Implementation of an efficient requirements-analysis supporting system using similarity measure techniques' *Information and Software Technology* 42: 429-438.

Prabhakar B K (1999) 'Internet based collaboration software: a study of impacts on distributed collaborative work' Thesis Dissertation [on-line accessed 17 Jul 2000] URL: www.lib.com/dissertations/previe_all/9930342

Saiden H and Dale R (2000) 'Requirements engineering: making the connection between the software developer and customer' *Information and Software Technology* 42: 419-428.

Sommerville I and Sawyer P (1997) *Requirements engineering – a good practice guide*, John Wiley & Sons, England.

Thanasankit T (1999) 'Social interpretation of evolving requirements – the influence of Thai culture', in Zowghi, D. (ed) (1999): *Proceedings of The Fourth Australian Conference on Requirements Engineering*, Macquarie University, Australia, 29-30 September, 1999, pp. 87-102.

Thanasankit T and Corbitt B J (1999) 'Towards understanding managing requirements engineering: a case study of a Thai

software house', in Yoong, P. (ed): *Proceedings of 10ᵗʰ Australasian Conference on Information Systems,* Vol. 2, 1-3 December, Victoria University of Wellington, New Zealand, pp. 993-1013.

Urquhart C (1996) 'Tell me what you want: a tale of an analyst, a client and the search for solutions - a case study in requirements gathering' *Proceedings of the 7th Australasian Conference on Information Systems*, Tasmania.

Viller S Bowers J and Rodden T (1999) 'Human factors in requirements engineering: a survey of human sciences literature relevant to the improvement of dependable systems development processes' *Interacting with Computers* 11: 665-698.

Related Content

Design Patterns Formal Composition and Analysis
Halima Douibiand Faiza Belala (2019). *International Journal of Information Technologies and Systems Approach (pp. 1-21).*
www.irma-international.org/article/design-patterns-formal-composition-and-analysis/230302

Comparing and Contrasting Rough Set with Logistic Regression for a Dataset
Renu Vashistand M. L. Garg (2014). *International Journal of Rough Sets and Data Analysis (pp. 81-98).*
www.irma-international.org/article/comparing-and-contrasting-rough-set-with-logistic-regression-for-a-dataset/111314

Innovation of Management Accounting Practices and Techniques
Davood Askarany (2015). *Encyclopedia of Information Science and Technology, Third Edition (pp. 10-19).*
www.irma-international.org/chapter/innovation-of-management-accounting-practices-and-techniques/112310

Rough-Set-Based Decision Model for Incomplete Information Systems
Safiye Turgay, Orhan Torkuland Tahsin Turgay (2018). *Encyclopedia of Information Science and Technology, Fourth Edition (pp. 2200-2212).*
www.irma-international.org/chapter/rough-set-based-decision-model-for-incomplete-information-systems/183932

Trend-Aware Data Imputation Based on Generative Adversarial Network for Time Series
Han Li, Zhenxiong Liu, Jixiang Niu, Zhongguo Yangand Sikandar Ali (2023). *International Journal of Information Technologies and Systems Approach (pp. 1-17).*
www.irma-international.org/article/trend-aware-data-imputation-based-on-generative-adversarial-network-for-time-series/325212