

CROSS-CULTURAL SOFTWARE ENGINEERING

INTERKULTURALNI SOFTVERSKI INŽENJERING

Hannu Jaakkola, Anneli Heimbürger

Tampere University of Technology, Pori, Finland; University of Jyväskylä, Finland
Tehnološko sveučilište Tampere, Pori, Finska; Sveučilište Jyväskylä, Finska

Abstract

One of the leading trends in software development is globalization. This trend can be seen from two different viewpoints: the software development in distributed cross-cultural organizations and the products distributed in the global market. The latter aspect means that developers must recognize the cultural differences of the users and produce adaptable, culturally sensitive products. This aspect will be left outside the scope of this paper. Instead, we will concentrate on analyzing the complexity caused by the globalization of software engineering work. Globalization is mostly motivated by economic reasons – a cheaper workforce is expected to reduce the costs of the work. However, in practice, it is not only a question of economics. Globalization has consequences in management culture, in distribution of the work, in ways of communication, and in many other aspects. In our paper we will open the discussion on the problems of the globalization of software development work. In spite of the importance of the topic only a few public studies on it are available. The paper analyzes the different organizational aspects of globalization and reviews experiences based on practical studies of the topic. The goal is to recognize the right kind of globalization path and also to point out the key issues worth recognizing when making the decision whether to globalize.

Sažetak

Jedan od vodećih trendova u razvoju softvera je globalizacija. Na ovaj trend možemo gledati s dva stajališta: razvoja softvera u distribuiranim interkulturalnim organizacijama i proizvoda distribuiranih na globalnom tržištu. Potonje gledište znači da sudionici u razvoju istih moraju prepoznati kulturološke razlike korisnika i razviti kulturno prilagodljive proizvode. To gledište, međutim, neće biti područjem ovog rada. Umjesto toga, usredotočit ćemo se na analizu složenosti uzrokovanu globalizacijom razvoja softvera. Globalizaciju uglavnom motivira ekonomizacija – očekuje se da jeftinija radna snaga snizi cijenu proizvodnje. Međutim, u praksi ovo nije samo pitanje ekonomiziranja. Globalizacija ima svoje posljedice u kulturi upravljanja, u raspodjeli rada, načinu komuniciranja i mnogim drugim aspektima. U svom uradku otvorit ćemo raspravu o problemima globalizacije razvoja softvera. Unatoč njenoj važnosti, postoji samo nekoliko dostupnih studija objavljenih o toj temi. Uradak analizira različite organizacijske aspekte globalizacije i osvrće se na iskustva temeljena na prekičnom proučavanju teme. Cilj je prepoznati pravu vrstu puta u globalizaciju kao i ukazati na važne probleme koje valja prepoznati pri donošenju odluke o globalizaciji.

1. Introduction

This paper is based on the on-going work in the authors' organization. The *UbiKnowS* (Ubiquitous Cross Cultural Knowledge Spaces) project – the main source of this paper - is financed by the Academy of Finland and JSPS (Japan) and is made in collaboration by both of the authors of this paper. Other collaborative partners are Keio University (Japan) and Komazawa University (Japan). The aim of this project is to study the characteristics of DCCI work:

- Distributed: Geographically diverse

- Cross-cultural: Members representing different cultural backgrounds
- Collaborative: Common goal, tasks part of the whole
- Intellectual: "Brain work"

The focus is on understanding (modeling) the processes, understanding the cultural differences, and developing a set of proper tools to support DCCI work. This project is sector independent, but has clear links to SE because of the background of the authors?. The other project, STEP (Steps in Software Business Globalization), studies the cross-

cultural aspects of SE work. The project is funded by the Finnish Funding Agency for Technology and Innovation (Tekes). The University of Jyväskylä (JYU) has been the organization responsible for the application processed in both the UbiKnows and STEP projects, and JYU is also the coordinator of the STEP project.

In addition, other research projects of the first author are linked to the topic: DDKM/SSMC (Seamless Solutions and Mobile Connectivity) is developing interoperability (SOA) solutions for information systems in distributed disaster knowledge management – including the aspects in cross-*cultural communication using models?*. Software Process Improvement studies (the EliTe project) cover aspects of Software Process Improvement; cultural aspects are elements of the *proscriptive process model* setting the target model of process improvement.

The goal of this paper is to provide the means to analyze the problems connected to globalization from different viewpoints. More generally it also opens discussion relevant in connection with *Culture-Aware Software Engineering - CASE* (see also Jaakkola et al. 2009a). The analysis covers the *organizational view* to recognize the right kind of globalization path, and the *cognitive view* to understand the aspects of human behavior in a cross-cultural context.

Globalization is one of the current trends in software engineering (SE) that are discussed in Section 2. The difficulty of SE is increasing along with the distribution of work. In this paper we will introduce a framework based on three *complexity factors*:

- distribution,
- culture, and
- ownership.

The *distribution factor* is based on the structure of the organization. The simplest case is work in an organization that operates in one office only. All employees have an opportunity for face-to-face communication and project management is based on traditional management disciplines. After establishing branch offices, the complexity caused by distribution must be taken into account. The distribution factor is discussed in Section 3 of this paper. Another dimension of difficulty in an organization is based on cultural diversity (*culture factor*). To act in a culturally homogenous organization is much easier than in a culturally heterogeneous organization. The term “culture” itself is also a complex concept – as discussed in Section 4 of this paper. The *ownership factor* analyzes the direction of cultural dominance in a culturally heterogeneous organization. The direction may be *outbound* (we own) or *inbound* (they own); the role of

this factor is discussed in Section 5. Section 6 reports the results of some public studies on the topic and draws some conclusions.

2. Characteristics of Software Engineering

According to EITO (2007), the total value of the global software market was projected at 238 billion EUR. On national level (Finland) the figures are as follows: software product business revenues (composed of the software product business and all related services) accounted for 1.52 billion EUR in 2007 /1/. However, this volume does not include the value of instruments, telecommunications equipment, etc., which include embedded software as a meaningful component of the product; this sector represents over 20% of Finnish exports. The total number of ICT companies in Finland is 8,800. They employ 46,000 employees; of which 33 000 are employed in software development. Software companies are reasonably small. 67% of companies employ less than 20 employees; 72% of companies have an annual revenue below 20 million EUR. The average revenue per employee is 100,000 EUR.

One of the trends changing the characteristics of Software Engineering is *globalization*. There are several reasons for this process. They can be analyzed from three different viewpoints: (1) the characteristics of the software *business* itself, (2) the properties of the software *products*, and (3) the software development *processes* – i.e. in software engineering itself. The characteristics connected to the *business sector* include:

- the ambition towards bigger business units either by acquisitions or mergers;
- networking and specialization;
- the need to operate (geographically) closer to the clients;
- the growing need for skilled personnel;
- the costs of the strategic business factors (work, office space, etc.);
- globalization as a path to growth (of business and company size / value).

The software business has changed a lot during the fifty-year history of commercial computers. From the original segment of use – i.e. the public and banking sector – information systems (IS) now play a *critical role* in society. The ratio of software value compared to hardware value has changed dramatically and during recent decades the software business has been one of the *fastest growing* sectors of business in the industrialized world. At the same time, it has changed from a “specialized business” to a “traditional” one and in a way we can say that it has become *mature* and follows the *same laws* as any other business. However, the SE

business still consists of two main categories: mass markets and customized markets. In mass markets, cultural factors can be taken into consideration as localization processes. In customized markets, it is necessary to consider cultural factors carefully, otherwise the business will fail.

The properties of the *software products* also have an effect on the business trends. Typically, a modern software product is

- based on *industrial* development methods (strategic level reuse – product lines, component factories, etc.); this item would be included in the following list discussing characteristics of software development processes;
- *modular* – the product consists of configurable and adjustable components;
- dominated by *architectures* (development platforms and predefined architectural principles);
- a part of a complex *systems of systems* (importance of interfacing and interoperability is emphasized);
- growing in *size* (because of the inbuilt overhead and because of its complexity);
- planned to satisfy the needs of a bigger and bigger *client base* – the growing importance of *user value* is dominant and the product is often just a part of the solution bought by the client (solution = product, services related to it, maintenance, etc. integrated in the daily business of the client);
- operated as a *service* instead of a purchase-based client-“owned” product (new concepts like SaaS - Software as a service, ASP - Application Service Provider, SOA – Service Oriented Architecture).

The character of the products is also changing from traditional ones (“hand-touchable”) to *embedded*; a growing amount of software is embedded in everyday processes and in the products we use in daily life; these are invisible to the users, and also mostly excluded from the software business itself in statistical categorizations.

The tradition in the *software development process* is plan-driven. The software development process follows the step-by-step process, having its roots in the traditional “Waterfall Model” (see e.g. Pfleeger and Atlee 2006, pp. 45-62) and its variants. The strict follow-up of this kind of *plan-driven development culture* expects the freezing of the requirements of the software product at a very early phase of the development, which makes the development project unable to react to requirement changes. It is also documentation-oriented because of the rules derived from the quality system adopted by the developer. Iterative and incremental modifications

of the waterfall model, as well as prototyping, are adopted to be more reactive to user requirement changes. As a continuation of the trend discussion, some development process trends are listed below:

- The two-dimensional iterative software development process – The *Rational Unified Process* (RUP) - was introduced by Kruchten /2/ and Jacobsson /3/ and is widely adopted in the SE industry;
- As an alternative to the strictly plan-oriented software development process, the new *Agile approach* has been adopted and has become popular especially in the development of small software products; it is based on RUP and puts the focus on short iterations and the ability to react to requirement changes. Agile methods implement the principles of the Agile manifesto (2009). Its practical implementations include Scrum and XP.
- Increasingly software is developed as a collaborative activity (either under one management or by individual developers as members of a community).

The character of SE has also changed a lot as a result of the tools that support the work. The processing capacity of the computers (or other computing devices) allows more freedom to the developers than earlier. In addition, the transmission capacity of networks supports the *distribution of work* – and also the existence of distributed information systems operating over public and private network infrastructures.

For readers who are particularly interested in ICT and SE trends, relevant reading includes the ICSE conference paper of Barry Boehm (2006) and the book of Endres and Rombach (2003). The authors of this paper have handled the topic in their earlier publications from the point of view of software architectures (Jaakkola 2007) and SE education (Jaakkola et al. 2008). The overview above provides important background knowledge and views worth understanding in connection with the topic of this paper – the distribution of SE and the role of cross-cultural aspects.

3. Distribution Factor

One of the topics commonly discussed as a part of IT sector development is internationalization – a more general term in this context would be globalization. *Internationalization* is mostly a question of access to the wider market – e.g. in Finland it means access to a 240 Billion EUR worldwide market as opposed to 1.5 billion EUR of local business opportunities. In this narrow interpretation globalization means (1) the opening of international markets and (2) the

internationalization of the products (culture-aware adaptable products). In most cases this is not an easy step, but entails a capital-intensive investment that is often bigger than the product development investment itself. In addition to opening up a wider scope of business, international-level support activities must be established. The company must

be prepared for much heavier competition than in the national market. Entry into an already existing and highly penetrated market is decelerated by cultural factors. In the case of a totally new product, naturally the situation is easier. Figure 1 /4/ introduces the typical path to build access to the international market.



Figure 1. The company internationalization process

Following the path of Figure 1 – starting from the very simple beginning (foreign inquiries) the level of internationalization (the term used by the source) grows and in the middle the flow splits into two branches – production and development in

a multicultural context. In a way we can interpret the internationalization of the business as a way towards globalization in its general meaning. In the introduction (Section 2) we already listed some drivers of this trend.

A. Outside	Broker	Broker Network	Broker Offshoring	Broker Outsourcing	Traditional Subcon-tracting
B. Inter-Organizational	Virtual Organization	Distributed Virtual Organization	Traditional Offshore	Traditional Outsourcing	NULL
C. Intra-Organizational	Traditional	Distributed Traditional	Concern based Offshore	NULL	NULL
	1. One site	2. Multiple sites	3. Offshore	4. Outsource	5. Subcon-tract

Figure 2. Distribution and globalization of work

The globalization has many manifestations. Figure 2 structures globalization in two dimensions: organizational (rows) and administrative (columns). *Intra-organizational* is a type of organization with a coordinated tier/matrix management. Inter-organizational management is co-operative and distributed – every organizational unit has its own management that dominates decision-making. *Outside* organization (looking from “our organization’s” direction) is something that our organization is not able to rule directly at all. The columns represent the level of distribution. *One-site* is a traditional “one-office” organization, in which the geographical vicinity supports immediate communication. *Multiple sites* increase the level of difficulty caused by the distribution of operative points of actions. *Offshore* is based on process connectivity – separate organizational units participate in a single business process while having their own responsibilities in the process. *Outsourcing* increases the independence of the operative units; collaboration is based on contracts. *Subcontracting* is based on external component purchases, usually based on competition.

In the case of the globalization of SE, the interesting area is separated outlined in red in the matrix. The organizational forms outside it represent virtual and broker organizations that also exist in software development (e.g. B-2 –computer game development or open source software development communities; brokers are integrators having no development activity of their own). In this paper we do not have the opportunity to go into detail about every organizational type (cell in the table); the table is provided as a framework to analyze the problems that must be solved in every organizational type.

4. Cultural Factor

The concept of culture is manifold. The PhD thesis of Liikamaa /5/ handles the role of culture in project work. In her thesis she separates (as a synthesis of several sources) three different cultures. *National culture* is more dominant in the behavior of individuals than the organizational culture. Therefore, cultural aspects like language, education, religion, beliefs, attitudes and social organization depend on the activities of the organization. *Organizational culture* includes habits adopted by the organization. *Work Culture* covers similarities in behavior, interaction, decision-making, organization structure, and goals. People who have adopted the same work culture are able to communicate and transfer knowledge better than people from different work cultures. The PhD thesis of Koskinen /6/ completes the list with professional culture and project culture. *Professional culture* has its roots in

education and adopted practices typical of certain professions. *Project culture* is a cross-section of organizational and professional culture. According to King’s /7/article, cultures can be considered at four levels: national cultures and organizational cultures, which are already mentioned above, and, in addition, *organizational subcultures* and *subunit cultures*. Duzi et al. /8/ extend King’s categorization with *team cultures*. A more detailed review of different aspects of the concept of “culture” is given by Jaakkola et al. /9/. In conclusion, it is easy to recognize the multidimensional character of culture and its effect on the behavior of individuals.

For the purposes of this paper we will concentrate on a deeper analysis of the cultural aspects relevant to the globalization of SE. There are two widely-used frameworks in this topic – the one developed by Hofstede /10/ and the one developed by Lewis /11/; these are introduced briefly below. All those who are working, for example, on international software engineering projects are involved - in addition to the subject of the SE project itself - in another kind of development process. Cultural competence /12/ is a developmental process that evolves step-by-step over an extended period. Both individuals and organizations are at various levels of awareness, knowledge, and skills on the cultural competence continuum. Cultural competence is about respecting cultural differences and similarities.

There are several studies for assessing cultures (Bijl 1995; De Mente 2001; Hofstede and Hofstede 2004; Lewis 1999). These studies consider relations between people, motivational orientation, orientation towards risks, definition of self and others, attitudes to time, working methods, communication protocols, and attitudes to environments. Hofstede’s framework for assessing cultures is one of the most widely-used frameworks /13/. Hofstede’s approach proposes a set of cultural dimensions along which dominant value systems can be ordered. The framework consists of five dimensions: individualism/collectivism, power distance, masculinity/femininity, uncertainty avoidance, and long-term orientation/short-term orientation (Table 1). All dimensions are generalizations and individuals may vary from their society’s descriptors.

Hofstede’s scores should not be taken literally /14/. They provide interesting background information because they show differences in answers between groups of respondents. Hofstede’s cultural dimensions can be considered as one of the general level frameworks for cross-cultural SE studies. Different value systems affect human thinking, feelings, and actions, and the behavior of

Table 1. Summary of cultural dimensions by Hofstede (2004)

Dimension	Description of the dimension
Individualism/ Collectivism	Individualism/Collectivism describes the extent to which a society emphasizes the individual or the group. Individualistic societies encourage their members to be independent and look out for themselves. Collectivistic societies emphasize the group's responsibility for each individual.
Power distance	Power distance describes the extent to which a society accepts that power is distributed unequally. When the power distance is high, individuals prefer little consultation between superiors and subordinates. When the power distance is low, individuals prefer consultative styles of leadership.
Masculinity/Femininity	Masculinity/Femininity refers to the values more likely to be held in a society. Masculine societies are characterized by an emphasis on money and things. Feminine cultures are characterized by concerns for relationships, nurturing, and quality of life.
Uncertainty avoidance	Uncertainty avoidance refers to the extent that individuals in a culture are comfortable (or uncomfortable) with unstructured situations. Societies with high uncertainty avoidance prefer stability, structure, and precise managerial direction. In low uncertainty avoidance societies, people are comfortable with ambiguity, unstructured situations, and broad managerial guidance.
Long-term/Short-term orientation	Long-term/Short-term orientation refers to the extent to which a culture programs its members to accept delayed gratification of their material, social, and emotional needs. Business people in long-term oriented cultures are accustomed to working toward building strong positions in their markets and do not expect immediate results. In short-term oriented cultures the "bottom line" (the results of the past month, quarter, or year) is a major concern. Control systems are focused on it and managers are constantly judged by it.

teams and organizations as well as the progress of different processes such as SE projects.

Lewis' Model of Culture focuses more on communication and interaction skills /15/. Cultural behavior is not something willy-nilly, accidental, or whimsical. On the contrary, it is the end product of millennia of collected wisdom, filtered and passed down through hundreds of generations and translated into hardened, undiscussable core beliefs, values, notions, and persistent action patterns. As such, a culture cannot be depicted satisfactorily at random or evaluated according to impressions or recent observations. It is a largely finite, predictable, and enduring phenomenon – the essential key to

survival for a nation or cultural group. Today a significant part of worldwide business is carried out by international teams. They may be temporary or permanent and tackle different tasks: product launches, setting up joint ventures, devising new strategies in sales and marketing, establishing new HR processes globally. International teams may be characterized by considerable cultural diversity – national, professional, corporate, individual. Though this diversity may be a source of strength, the team will need to find solutions to the challenges of internal integration before it can solve ones of external adaptation - which may be to collaborate with other teams.

Table 2. Culture classification by Lewis (1999)

Linear Active	Multi-Active	Reactive
Talks half the time	Talks most of the time	Listens most of the time
Does one thing at a time	Does several things at once	Reacts to partner's action
Plans ahead step by step	Plans grand outline only	Looks at general principles
Polite but direct	Emotional Polite	Indirect
Partly conceals feelings	Displays feelings	Conceals feelings
Confronts with logic	Confronts emotionally	Never confronts
Dislikes losing face	Has good excuses	Must not lose face
Rarely interrupts	Often interrupts	Doesn't interrupt
Job-orientated	People-orientated	Very people-orientated
Sticks to facts	Feelings before facts	Statements are promises
Truth before diplomacy	Flexible truth	Diplomacy over truth

According to the Lewis Model of Culture, cultures can be classified into three main categories as described in Table 2. Project managers and team members in linear-active cultures generally demonstrate task orientation. They look for technical competence, place facts before sentiment, logic before emotion; they are deal-orientated, focusing their own attention and that of their staff/team/individuals on immediate achievements and results. They are orderly, stick to agendas and inspire staff with their careful planning. Multi-active project managers and team members are much more extrovert, rely on their eloquence and ability to persuade and use human force as an inspirational factor. They often complete human transactions emotionally, investing the time to developing the contact to the limit. Such project managers and team members are great networkers, working according to people-time rather than clock-time. Project managers and team members in reactive cultures are equally people-orientated but dominate with knowledge, patience, and quiet control. They display modesty and courtesy, despite their accepted seniority. They create a harmonious atmosphere for teamwork. Subtle body language replaces excessive words. They know their companies well (having spent years going round the various departments), giving them balance and the ability to react to a web of pressures. They are also paternalistic.

In a world of rapidly globalizing business, Internet electronic proximity and politico-economic association (EU, NAFTA, ASEAN, etc.), the ability to interact successfully with foreign partners in the spheres of commercial activity, diplomatic intercourse and scientific interchange is seen as increasingly essential and desirable. The Hofstede and Lewis models give us an interesting macro level framework to study cultural factors. However, in modeling, designing and implementing cross-cultural knowledge into software engineering development work processes, products, services and applications, we need more detailed contextual analysis i.e. application, situational, task and user specific analysis.

5. Ownership factor

The third factor introduced in section 1 (Introduction) is the *ownership* of the organization. In relation to cultural differences it is question of dominance – the direction of the cultural expectations in the organization. In this context we have to look at all cultural dimensions discussed in section 4. In different forms of distributed organizations, there are two ways to adapt to cultural differences: (1) allow all operative units to apply their own culture

(*cultural dissimilarity*), (2) unify the network to follow the same culture (*cultural similarity*). There is evidence of both approaches in connection with enterprise acquisitions and mergers.

The cultural similarity / dissimilarity strategy can be recognized in the analysis of all cultural categories discussed in the the first paragraphs of Section 4). In principle, it is a question of the permissive / non-permissive culture in the organization. In a *permissive multicultural organization*, the freedom to follow the existing cultural values exists. In most cases, the demand for similarity is seen to demand too many resources (e.g. unification of information systems and processes); alternatively, a multicultural environment might be experienced as a strength of the organization – different cultures have the best fit in different activities. The *non-permissive* approach expects organizational unity and does not accept diverse cultures adopted at any cultural level. There is evidence of both cultures; a short report on culture adaptation is given by Jaakkola et al. /16/ in the context of information system integration in merging companies.

In the discussion of the ownership factor, the difference in national cultures is highlighted more than the other cultural differences. The starting point for discussion is that the “owner talks” – i.e. the culture of the company / process / product dominates the unification of culture. However, the situation is different in a case where (1) an Indian company buys a Finnish software company (Wipro – Saraware) than in a case where (2) a Finnish company extends its operations to China (Nokia, Tieto – both establishing a local company) or to the Czech Republic (Tieto – offshoring, also the acquisition of a Czech company). The success of cultural unification is more successful in the case of cultural closeness than in the case of cultures far from each other. In our research projects (Section 1) we will concentrate on an in-depth analysis of the path of some globalization decisions. The motivation for globalization differs from case to case; the decision is not simply an economic one, because the growth of operative complexity will cause overhead expenses in the cost structure and the cost differences between countries tend to melt away in a reasonably short time. For example, the cost of a software specialist in China (in a Finnish company; approx. 3,500 Euro) is very close to the price of Finnish employee in Finland (approx. 4,000–5,000 Euro).

6. Conclusions

In our paper we have introduced a framework to support the understanding of the problems of a multicultural distributed organization. The

framework is general, although the authors have background experience from the software industry. The components of the framework model are *organizational distribution, diversity of cultures and direction of the ownership*. In the analysis the factors derived from all of these are interacting and relevant items to support decision-making whether to globalize or not. The motivation to globalize also varies and the reasons behind the decision-making vary from case to case. In the organization of work, the important factor common to all the cases seem to be success in communication (face-to-face vs. distance), division of work (what kind of tasks), and management culture at different levels (feedback, attitude in time, etc.).

There is a lack of objective studies of the success factors of globalization in the SE industry. In connection with our research projects, the following are recommended for further reading. The article of *Walsham* (2002) provides a theoretical framework for cross-cultural software production and use. In their article, *Siakas et al.* (1999) deal with the attitudes to software quality and Total Quality Management (TQM) in the organization. The framework applied is Hofstede's five-dimension model. The same framework is also applied by *Abraham* (2009) in his conference paper analyzing cultural differences as a part of software life cycle management. The conference paper of *Hawthorne and Perry* (2005) opens discussion on SE education in distributed and multicultural organizations. The same topic is also handled by Jaakkola et al (2009a). One of the reports providing very concrete results of SE outsourcing has been published by Krishna and Walsham (2004). They report experiences in outsourced software development in India, Japan and in some European countries. The same topic is discussed by *Borchers* (2003) in his conference paper – the nationalities discussed are Japanese, Indian, and American. This analysis also applies Hofstede's indices. A paper by *Simcock* (1998) reports a case of the cultural mix in SE design teams as a part of a project included in a university-level IT curriculum.

Our work in this topic started some months ago. The goal is to produce applicable knowledge for the SE industry to support their decision-making on globalization. The aim is to transfer the decision-making from the *expectation level to realism*; too often globalization decisions are not based on facts and the decision-makers are not fully informed on the problems appearing and the costs caused after implementing the decision to globalize. The fact is, in any case, that the SE industry as well as other industries is a part of the globalizing world, in which employees move, and companies network to collaborate. In many cases, the products are also

connected to a whole, in which the final system is a configuration of subsystems delivered by different organizations.

References

- /1/ Rönkkö Mikko et al., National Software Industry Survey 2008: The Finnish Software Industry in 2007. Helsinki University of Technology Report. ISBN: 978-951-22-9736-8.
- /2/ Kruchten Philippe (ed.). (1998). *The Rational Unified Process: An Introduction*. Addison-Wesley
- /3/ Jacobson I., Booch G., Rumbaugh J. (1999). *The Unified Software Development Process*. Addison-Wesley.
- /4/ Mäkilouko M.I. (2001), *Leading Multinational Project Teams: Formal, Country Specific Perspective*. Tampere University of technology, Publications 343, 2001.
- /5/ Liikamaa, K., Piilevätietoja projektipäällikönten kompetenssit. PhD Thesis. Tampere University of Technology. Publication 628. Tampere. 2006. In Finnish.
- /6/ Koskinen, Kaj U., *Management of Tacit Knowledge in a Project Work Context*. Acta Polytechnica Scandinavica. Industrial Engineering and Management Series No 10. Espoo. 2001.
- /7/ King, W. R. 2007. *A Research Agenda for the Relationships between Culture and Knowledge Management*. Knowledge and Process Management, Vol. 14, no. 3, pp. 226 – 236.
- /8/ Duzi, M., Heimbürger, A., Tokuda, T., Vojtas, P. and Yoshida, N. *Multi-Agent Knowledge Modelling*. in Kiyoki, Y., Tokuda, T. and Jaakkola, H. eds. *Information Modelling and Knowledge Bases XX*. , IOS Press, Amsterdam, 2009. pp. 411-428.
- /9/ Jaakkola, H., Heimbürger, A. and Linna, P. (2009a). *Knowledge Oriented Software Engineering Process in Multi-Cultural Context*. *Software Quality Journal* (Springer). Manuscript. Submitted for review.
- /10/ Hofstede, G. and Hofstede, G. J. 2004. *Cultures and Organizations: Software of the Mind: Intercultural Cooperation and Its Importance for Survival*. New York: McGraw-Hill. 300 p.
- /11/ Lewis, R. D. 1999. *When Cultures Collide. Managing Successfully Across Cultures*. London: Nicholas Brealey Publishing. 462 p.
- /12/ Ibidem
- /13/ Hofstede, G. and Hofstede, G. J. 2004. *Cultures and Organizations: Software of the Mind: Intercultural Cooperation and Its Importance for Survival*. New York: McGraw-Hill.
- /14/ Hofstede, G. 2003. *Geert Hofstede™ Cultural Dimensions*. <http://www.geert-hofstede.com/>. Accessed 20 March, 2009.
- /15/ Lewis, R. D. 1999. *When Cultures Collide. Managing Successfully Across Cultures*. London: Nicholas Brealey Publishing.
- /16/ Jaakkola, H., Uusitalo, O. and Henno, J. (2009b). *Adaptation in IS Integration and Enterprise Migration*. Cincin-Sain, M. ed. MIPRO 2009, Opatija, Croatia.

Literature

1. Abraham, L. R. 2009. *India Software Engineering Conference*. Proceeding of the 2nd Annual Conference on India Software Engineering Conference, pp. 95-100. ACM New York, NY, USA
2. Agile (2009). *Agile Manifesto*. <http://agilemanifesto.org/>. Accessed 14 April 2009.

3. Boehm, B. (2006). A view of 20th and 21st Century Software Engineering Proceedings of the 28th International Conference on Software Engineering (ICSE), ACM SIGSOFT and IEEE TCSE, Shanghai, China.
4. Borcheres Greg (2003). The software engineering impacts of cultural factors on multi-cultural software development teams. Proceedings of the 25th International Conference on Software Engineering (ICSE 2003). IEEE.
5. EITO - European Information Technology Observatory (2007). Eito 2007 Report. Technical report, EITO.
6. Endres, A. and Rombach, D. (2003). A Handbook of Software and Systems Engineering - Empirical Observations, Laws and Theories. Pearson and Addison Wesley, 2003.
7. Hawthorne, M. J. and Perry, D. E. 2005. Software Engineering Education in the Era of Outsourcing, Distributed Development, and Open Source Software: Challenges and Opportunities", International Conference on Software Engineering (ICSE2005), St. Louis MO, May 2005.
8. Jaakkola, H. (2007). Software Architectures and the Architecture of Long-Distance Knowledge Sharing, Analysis and Delivery Platform. Invited paper. First International Symposium on Universal Communication (ISUC), NICT, Kyoto, Japan.
9. Jaakkola, H. and Henno, J. (2008). Trends in Software Design Education. Cincin-Sain, M. ed. MIPRO 2008. 31st International Convention on Information and Communication Technology, Electronics and Microelectronics - Proceedings vol. IV, Opatija, Croatia, 2008, 89 -93.
10. Jaakkola, H., Heimbürger, A. and Henno, J. (2009). The Roles of Knowledge and Context in Context-Aware Software Engineering - in Terms of Education and Communication Cincin-Sain, M. ed. MIPRO 2009, MIPRO and IEEE, Opatija, Croatia, 2009.
11. Krishna S., S.S., Walsham G. (2004). Managing Cross-Cultural Issues in Global Software Outsourcing. Communications of the ACM, 47 (4). 62-66.
12. Kuitunen et al., Finnish Software Product Business: Results from the National Software Industry Survey 2005. TEKES & Centre of Expertise for Software Product Business.
13. Pfleeger, S.L. and Atlee, J.M. Software Engineering - Theory and Practice. Pearson and Prentice Hall, 2006.
14. Siakas, K., Georgiadou, E., Sadler, C. 1999. Software Quality Management from a Cross-Cultural Viewpoint. Software Quality Journal, Volume 8, Number 2, October 1999, pp. 85-95.
15. Simcock Alec J. (1998). Does a multicultural mix bring extra dimensions to software engineering design teams. Global Journal of Engineering Education 2 (3), pp. 263-270.
16. Walsham, G. 2002. Cross-cultural software production and use: a structural analysis. MIS Quarterly.