Enhancing Agile Methods for Multi-cultural Software Project Teams

A Sutharshan
School of Computer Science and Security Science, Edith Cowan University
2 Bradford Street, Mount Lawley, Western Australia, 6050, Australia
Tel: 61-8-9310-1164 E-mail: anu.sutharshan@transport.wa.gov.au

S P Maj
School of Computer Science and Security Science, Edith Cowan University
2 Bradford Street, Mount Lawley, Western Australia, 6050, Australia

Abstract

It is well documented that software projects are typically over schedule, over budget and often do not meet user requirements. The main problems are all associated with people related issues. In order to address this problem the Agile philosophy was introduced with an associated portfolio of Agile methods. These methods are specifically designed to improve software project team management. However it is now increasingly common for software projects to have multicultural team members. It is well documented that people from different cultures have considerably different expectations and methods of interacting in a team environment. In order to address this problem cultural specific Agile attributes were defined based on Hofstede's cultural dimensions. The result of this study gives an insight to how cultural differences may affect a software methodology implementation, specifically Agile and how these problems can be addressed. Hence it is possible to select appropriate 'culture and Agile specific attributes' when working with multicultural software project team to help software development projects with agile methods.

Keywords: Hofstede, Agile, Software project management, Multi cultural team

1. The Software Engineering Crisis

Software development is an expensive, and often a difficult process (Cerpa & Verner, 2009; Imamoglu & Gozlu, 2008). Although corporate expenditure on information technology (IT) has dropped in recent years, firms spend more than a trillion US dollars a year on IT (Love, Ghoneim, & Irani, 2004). It is well documented that the majority of software project are problematic. A survey of over 8000 projects undertaken in the year 2000 by 350 US companies revealed that one third of the projects were never completed and one half succeeded only partially, that is, with partial functionalities, major cost overruns, and significant delays (Lamsweerde, 2000). Software project management continues to be a challenging area for practitioners: more than half of all software projects experience severe difficulties and/or failure (Standish Group, 2004). The Standish Group's "CHAOS Report," (Eveleens & Verhoef, 2010; Standish Group, 2004), a widely respected survey of software projects in industry and government, estimated that, in the year 2009, only 32% of software projects in large enterprises succeeded (i.e., produced acceptable results that were delivered close to on-time and on-budget). 44% were "challenged" (significantly over budget and schedule), and 24% failed to deliver any usable result (Table 1). The projects that are in trouble have an average budget overrun of 56%. This represents a serious and chronic problem.

In order to address this on-going problem many different software development methods have been developed and introduced. There are three broad categories of methods: formal, structured and soft. Formal methods, such as Z and the Vienna Definition Method (VDM), are mathematically based allowed proof of completeness and consistency (Abernethy, Kelly, Sobel, Kiper, & Powell, 2000). However such methods are complex and beyond the scope of most system developers. Structured methods, such as Structured Systems Analysis Design Method (SSADM), provide detailed guidelines along with commonly used tools and techniques such as Data Flow Diagrams (DFDs) (Cohen & Bar-On, 1989). Soft methods such as Soft Systems Methodology (SSM), Multiview and Effective Technical and Human Implementation of Computer-Based Systems (ETHICS), were designed to accommodate the problems associated with team dynamics (Bustard & Lundy, 1995). It is now recognized that human factors are the most significant components that determine project success.

Clearly, personalities are complex and many software engineers exhibit a variety of different traits and attitudes (Bostrom & Kaiser, 1981). The importance of end user participation is also an important factor in project success (Hartwick & Barki, 1994). Thimbleby (2001) has stated that, "the lack of user centered design is the classic

reason for the failure of almost all programmed systems". There is considerable research summarizing supporting user participation and involvement in IT (Hartwick & Barki, 1994). Hartwick & Barki further stressed the importance of the IT organization and the user community working together to develop high value systems. The need for the participation and involvement of end users in IT development has been recognized (Avital & Vandenbosch, 2000). The philosophy is that products have no reason to exist without people, therefore people and their goals must be a critical part of the process and the practice (Rohn et al., 2002). Hence a range of different software development methods were developed and introduced such as Extreme Programming (XP), Lean Development, Feature Driven Development etc. In 2000 the Agile philosophy was introduced as framework of principles for software development projects. The need for the participation and involvement of users in IT development was recognized even in 70s (Lucas, 1971). Human related skills became important as a result of increased user involvement in the IS development process (Cheney, 1988). Cheney (1988) also identified the changing emphasis towards general interpersonal skills and, specifically, the ability to communicate with end users involved in the IS development process.

2. Agile Philosophy

Agile is a framework of principles that employs a range of different software methods – referred to as Agile methods. Software through people is the motto of the Agile Manifesto (Highsmith, 2002). The Agile philosophy is primarily informal with minimal documentation. Hence the emphasis is on verbal and social communication within the development team (Valencia, Olivera, & Sim, 2007). Significantly Agile based methods have been gaining widespread acceptance amongst practitioners (Valencia et al., 2007).

The Agile philosophy evolved based on four key values:

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan

The principles behind this manifesto recommend that change should be welcomed at every stage of the software development cycle; that working software should be delivered frequently; and that conveying information via face-to-face conversation is more efficient than through written documentation (Valencia et al., 2007). A 2003 global survey of experience using Agile methodologies carried out by an Australian company produced the results that have been summarized below:

- 88% of organizations cited improved productivity
- 84% of organizations reported improved quality of software products
- 46% of respondents reported that development costs were unchanged using Agile methodologies, while 49% stated that costs were reduced or significantly reduced
- 83% stated that business satisfaction was higher or significantly higher
- 48% cited that the most positive feature of Agile methodologies was their ability to 'respond to change rather than follow a predefined plan' (Shine Technologies, 2003).

Results from a survey done in 2006 at Microsoft to identify what the participants thought were the top 10 benefits with Agile development are listed in table 2 (Begel & Nagappan, 2007). The top benefit was improved communication and coordination among team members. The second most cited benefit was timely and rapid software releases. For a good software project to be successful, it has been indicated that focus should be placed on the processes, technology and people in order to achieve better performance, and the people-focus is by far the component that gets the least attention (Leonard, 2002). However it is now common for projects to consist of team member from very diverse cultures – each with their own unique expectations and communication methods.

3. Multicultural Software Development Teams

Cultural factors may have an impact on the success of software development. This problem may be exacerbated when projects are multicultural. Culture is one area of social science that receives constant attention (Jones, 2007). It has been reported that until studies were done on cultural factors, there was difficulty in understanding software development problems with two projects that involved software developers from India, Japan and the United States (Chand, 2004). Cultural study and inter-cultural communication have been of interest since 1950 (Rogers, Hart, & Mike, 2002). Much interest has been placed on culture in business in the last two decades, and

it has never been as important in business terms as it is today. Culture is important for many aspects of business life especially when a business must interface with people, either as customers, employees, suppliers or stakeholders.

Cross-cultural research has had most value therefore when it has been able to provide substance to modern management practices and techniques. Differences between nations and societies make it critical to understand how institutional and cultural factors influence IT application (Martinsons & Davison, 1998). Theories tend to be developed in a specific cultural environment and expect to transfer seamlessly to a different cultural environment is naïve (Martinsons & Davison, 1998). The study of the field began in earnest with the work of Hofstede with his landmark study of IBM (Hofstede, 1980).

4. Hofsteds Cultural Dimensions

Hofstede's work on culture is the most widely cited in cultural studies (Hofstede, 1997; Jones, 2007). His results are based on an extensive study consisting of 116,000 questionnaires, from which over 60,000 people responded from over 50 countries. From the data he obtained he provided a factor analysis of 32 questions in 40 countries. From this he identified four dimensions of societal culture (Hofstede, 1980) - power distance index (PDI), Individualism (IDV), Masculinity (MAS), Uncertainty avoidance index (UAI). A subsequent study revealed a fifth dimension - Long term orientation (LTO).

According to Hofstede (1997) the most important differences between cultures can be captured by finding out to what extent members of these cultures differ with regard to five values:

- Power distance index (mainly the degree of dependence between boss and subordinates),

- Individualism (the degree in which everyone is expected to look after him or herself) versus Collectivism (integration into cohesive in-groups),

- Masculinity versus femininity (masculinity is high in societies in which social gender roles are clearly distinct and femininity is high in societies in which social gender roles overlap),

- Uncertainty avoidance index (the extent to which the members of a culture feel threatened by uncertain or unknown situations)

- Long term orientation (the importance attached to the future versus the past and present.

Figure 1 shows Hofstede's values for these cultural dimensions for Australia, India, United Kingdom and Japan and a clear picture indicating that cultures around the world have different patterns of social behaviour and interaction. From the figure it is clear that what works for one culture may or may not work for somewhere else.

5. Cross-Cultural Research in Implementing Agile

In recent years multi-cultural practices and values have become significantly conspicuous in corporate businesses (Kanungo, 2006). According to Herbsleb (2007), globally-distributed projects are rapidly becoming the norm for large software systems, even as it becomes clear that global distribution of a project seriously impairs critical coordination mechanisms. Over decades, organizations are devoting effort to address this issue. According to Rama Prasad Kanungo (2006) openness, transparency, acceptance of ideas and products, willingness to engage are all some of the shared values that will reflect interdependence between cultures. Cross cultural research has had most value when it has been able to provide substance to modern management practices and techniques (Jones, 2007).

There are and will continue to be inter-cultural factors that affect both collocated and distributed software development efforts (MacGregor, Hsieh, & Kruchten, 2005). Connections between Software Development Methodologies (SDM) and cultural issues have been discussed previously (Abrahamsson, Salo, Ronkainen, & Warsta, 2002; Yourdon, 1986). There is clear visibility of connections between SDMs and cultural related issues on the national level. It is no longer unusual for a large software project to have teams in more than one location, often in more than one continent. Many forces have conspired to bring about this situation, including concern for cost, the need to tap global pools to acquire highly skilled resources, finding an appropriate mix of expertise for a project, satisfying investment requirements imposed by governments in foreign markets, and mergers and acquisitions (Herbsleb, 2007). There is little reason to expect these factors to diminish in the future. Rather, it appears that we face increasing globalization of markets and production, increasing the pressure to distribute projects globally.

This paper emphasizes or studies culture and software methodology, specifically Agile methodology. The authors are addressing problems that arise when developers from different culture work together and also focus on problems that originate in gaps between a national culture and the culture that is inspired by a given SDM,

here Agile methodology. This research deals with connections between cultural characteristics and the willingness of software engineering teams to adopt a given SDM. As it is well known software development entails many problems including clashes in customer-developer communication, bugged software, misunderstanding among team members, requirements not being clearly understood by the team etc.

In order to address these multicultural software development concerns the authors analysed the Agile principles (defined by Agile Manifesto) for cross-cultural factors. Agile specific cultural attributes relevant to multicultural concerns were identified (Table 3). A good relationship pattern was clearly seen and all these Agile specific cultural attributes corresponded to Hofstede's cultural dimensions. Further to this, the culture and Agile specific attributes were categorized into different groups based on Hofstede's cultural dimensions (Table 4). Hofstede's previous study helped in identifying societal cultural dimensions for different nationalities. Based on these values, a set of cultural specific attributes were matched (Table 5).

For example, Power Distance Index for India is high with a value of 77, and in turn Power Distance Index (PDI) for Australia is low with a value of 36 and world average value is 55. Power Distance Index dimension looks at how much a culture does or does not value hierarchical relationships and respect for authority (Hofstede, 1998). The set of cultural specific Agile attributes that have relationship with PDI include:

- Trust people more than process
- Transparency
- Authoritative
- Quick Decision Making
- Empowered
- Proactiveness
- Management support
- Collective ownership
- Blame sharing
- Negotiation
- Conflict Resolution

A mixed team with different cultural dimensions will be highly problematic. This condition is not necessarily forced upon the population, but rather accepted by the society as their cultural heritages. Cultures with a lower power distance have employees who are not afraid of authority and the bosses are not autocratic or paternalistic. Employees from these cultures express a preference for a consultative style of decision making and participative management style. In turn cultures with a high power distance are not expected to be involved in decision making and participative. Thus an understanding of the culture specific Agile attributes will help in a better team management and a successful project team. The cultural bias can be used to build a sense of team management and in turn a better software development project. Previous research in similar fields is listed in Table 6. The findings of this research aim to make contribution to agile adoption and to societal culture research. In addition this research contributes to and extends theoretical knowledge on agile adoption process. This study is the first to research agile methodology implementation in different cultures.

6. Conclusion

It is difficult to get every team member to change how they think overnight. Rather than trying to build understanding about agile methods as a full set, this study helps select and introduce attributes that best apply to a given societal culture. Specifically, based on our research on cultural differences and issues related to software development teams, a model is presented to help understand what attributes to consider depending on the national cultural dimensions. This will help to adapt and adjust the team expectations to the realities of the cultural impacts. The outcome of this study is to provide a framework that can be used to describe, analyse, and change culture that will help to implement Agile methods. We hope that the results of our efforts will be not only tangible and utilizable set of results but also a clear path for future research.

References

Abernethy, K., Kelly, J., Sobel, A., Kiper, J. D., & Powell, J. (2000). *Technology Transfer Issues for Formal Methods of Software Specification.* Paper presented at the 13th Conference on Software Engineering Education and Training, USA.

Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2002). *Agile software development methods - Review and Analysis*: University of Oulu.

Avital, M., & Vandenbosch, B. (2000). *The relationship between psychological ownership and IT-driven value.* Paper presented at the Proceedings of the twenty first international conference on Information systems Brisbane, Queensland, Australia

Begel, A., & Nagappan, N. (2007). *Usage and Perceptions of Agile Software Development in an Industrial Context: An Exploratory Study.* Paper presented at the First International Symposium on Empirical Software Engineering and Management.

Bostrom, R. P., & Kaiser, K. M. (1981). *Personality Differences Within Systems Project Teams: Implications for Designing Solving Centres*.

Bustard, D. W., & Lundy, P. J. (1995). *Enhancing soft systems analysis with formal modelling*. Paper presented at the RE '95 Proceedings of the Second IEEE International Symposium on Requirements Engineering.

Cerpa, N., & Verner, J. M. (2009). Why did your project fail? *Communications of the ACM, 52*(12), 130 - 134.

Chand, D. (2004). Is an understanding of national cultures essential for Global IT Managers?

Cheney, P. H. (1988). *Information Systems Skills Requirements: 1980 & 1988*.

Cohen, I., & Bar-On, D. (1989). *Working with Structured Methods - a Case Study*. Paper presented at the Proceedings of fourth conference on Computer Systems and Software Engineering.

Eveleens, L., & Verhoef, C. (2010). The Rise and Fall of the Chaos Report Figures. *IEEE software*.

Hartwick, J., & Barki, H. (1994). Explaining the role of User participation in Information System use. *Management Science, 40*(4), 440 - 465.

Herbsleb, J. D. (2007). *Global Software Engineering: The Future of Socia-technical Coordination.* Paper presented at the Future of Software Engineering.

Highsmith, J. (2002). *Agile Software Development Ecosystems*. Indianapolis: Addison-Wesley.

Hofstede, G. (1980). *Culture's consequences: International differences in work-related values*. Newbury Park, CA: Sage.

Hofstede, G. (1997). *Culture and Organisations: Software of the Mind*. New York: McGraw-Hill.

Hofstede, G. (1998). Attitudes, Values and Organisational Culture: Disentangling the concepts. *Organisational studies, 19*(3), 477.

Imamoglu, O., & Gozlu, S. (2008). *The Sources of Success and Failure of Information Technology Projects: Project Manager's Perspective*. Paper presented at the PICMET 2008 Proceedings.

Jones, M. L. (2007). *Hofstede - Culturally questionable?* : University of Wollongong.

Kanungo, R. P. (2006). Cross culture and business practice: are they coterminous or cross-verging? . *Emarald, 13*(1), 23-31.

Lamsweerde, A. v. (2000). *Requirements engineering in the year 00: a research perspective*. Paper presented at the Proceedings of the 22nd international conference on Software engineering.

Leonard, A. (2002). *Enabling End Users To Be More Efficient During Systems Development.* Paper presented at the Proceedings of SAICSIT 2002.

Love, P. E. D., Ghoneim, A., & Irani, Z. (2004). Information technology evaluation: classifying indirect costs using the structured case method. *Journal of Enterprise Information Management, 17*(4), 312-325.

Lucas, H. C. (1971). *A User-Oriented Approach to Systems Design*. Paper presented at the Proceedings of the 1971 annual conference.

MacGregor, E., Hsieh, Y., & Kruchten, P. (2005). *Cultural Patterns in Software Process Mishaps: Incidents in Global Projects.* Paper presented at the Human and Social Factors of Software Engineering (HSSE), Missouri, USA.

Martinsons, M. G., & Davison, R. M. (1998). Cultural Considerations in Business Process Change.

Rogers, E. M., Hart, W. B., & Mike, Y. (2002). Edward T Hall and the history of Intercultural Communication: The United States and Japan. *Keio Communication Review, 24*.

Rohn, J. A., Spool, J., Ektare, M., Koyani, S., Muller, M. J., & Redish, J. G. (2002). *Usability in practice*

alternatives to formative evaluations - evolution and revolution. Paper presented at the Conference on human factors and computing systems.

Shine Technologies. (2003). Agile methodologies - Survey results. Retrieved 27th May 2010, 2010, from http://www.shinetech.com/attachments/104_ShineTechAgileSurvey2003-01-17.pdf

Standish Group. (2004). *2004 Third Quarter Research Report*. West Yarmouth, MA, USA: The Standish Group International.

Valencia, R. E. G., Olivera, V., & Sim, S. E. (2007). *Are Use Cases Beneficial for Developers Using Agile Requirements?* Paper presented at the Fifth International Workshop on Comparative Evaluation in Requirements Engineering.

Yourdon, E. (1986). *What ever happend to structured analysis?*

Table 1. Standish Chaos report – project benchmark over the years (Eveleens & Verhoef, 2010)

Year	**Successful (%)**	**Challenged (%)**	**Failed (%)**
1994	16	52	31
1996	27	33	40
1998	26	46	28
2000	28	49	23
2004	29	53	18
2006	35	46	19
2009	32	44	24

Table 2. Benefits to Agile development methodologies (Begel & Nagappan, 2007)

No.	**Benefits with Agile development**	**Participant number**
1.	Improved communications	121
2.	Quick releases	101
3.	Flexibility of design	86
4.	More reasonable process	65
5.	Increased quality	62
7.	Better customer focus	50
8.	Increased productivity	28
9.	Better morale	23
10.	Testing first	22

Table 3. Culture and Agile specific attributes defined based on Agile principles

Defined/Identified Culture and Agile Specific Attributes	**Agile Principles – defined by Agile Manifesto**											
	Satisfy the customer through early and continuous delivery of valuable software	Sustainable development is promoted, facilitating indefinite development	Simplicity is essential – the art of maximising the amount of work not done is essential	Welcome changing requirements, even late in development	Deliver working software frequently	Working software is the primary measure of progress	Continuous attention to technical excellence	Business people and developers must work together daily	Face-to-face communication is the best method of conveying information	The team regularly reflects on how to become more productive and efficient	The best work emerges from self-organising teams	Build projects around motivated individuals
Trust people more than process		✓			✓			✓		✓	✓	✓
Transparency	✓		✓		✓			✓	✓	✓	✓	✓
Team collaboration		✓			✓			✓		✓	✓	✓
Self-organizing team		✓			✓	✓	✓	✓		✓	✓	✓
Dedicated team		✓			✓			✓		✓	✓	✓
Risk Taking				✓							✓	
Innovation				✓						✓		
Authoritative					✓							
Quick Decision Making	✓	✓	✓	✓	✓	✓		✓		✓	✓	
Open and honest communication		✓			✓	✓			✓			
Tolerance for change	✓			✓	✓	✓						
Empowered												✓
Meeting deadlines and expectations	✓	✓		✓	✓	✓						
Proactiveness	✓	✓	✓		✓	✓						
Time keeping	✓	✓			✓							
Direct customer involvement	✓	✓		✓		✓			✓			
Management support	✓	✓			✓	✓		✓				✓
Collective ownership	✓	✓		✓				✓			✓	✓
Blame Sharing	✓			✓				✓			✓	
Negotiation	✓	✓		✓				✓		✓	✓	
Conflict Resolution	✓			✓				✓		✓	✓	

Table 4. Culture and Agile specific attributes versus Hofstede's cultural dimensions

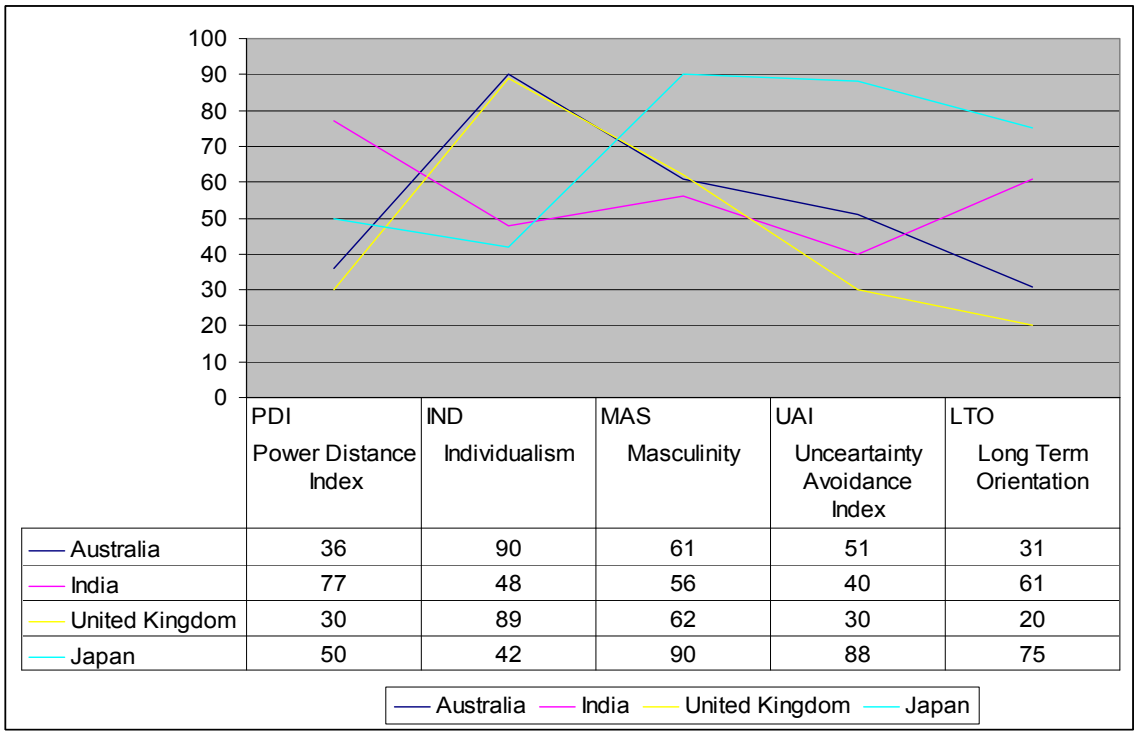
Culture and Agile specific attribute	**Hofstede Cultural Dimension**				
	Power Distance	Individualism	Masculinity	Uncertainty Avoidance Index	Long term Orientation
Trust people more than process	✓				✓
Transparency	✓				
Team collaboration		✓			
Self-organizing team		✓			
Dedicated team		✓			
Risk Taking				✓	✓
Innovation				✓	
Authoritative	✓		✓		
Quick Decision Making	✓	✓			✓
Open and Honest Communication		✓			
Tolerance for change				✓	
Empowered	✓		✓		
Meeting deadlines and expectations				✓	✓
Proactiveness	✓	✓			
Time keeping				✓	✓
Direct customer involvement		✓			
Management support	✓				
Collective ownership	✓	✓			
Blame Sharing	✓	✓			
Negotiation	✓	✓	✓		
Conflict Resolution	✓	✓	✓	✓	

Table 5. Hofstede's Cultural dimensions and related culture and Agile specific attributes

Hofstede's Cultural Dimensions	**Culture and Agile Specific Attributes**
Power Distance Index (PDI)	Trust people more than process
	Transparency
	Authoritative
	Quick Decision making
	Empowered
	Proactiveness
	Management support
	Collective ownership
	Blame Sharing
	Negotiation
	Conflict Resolution
Individualism (IND)	Team Collaboration
	Self organising Team
	Dedicated Team
	Quick Decision making
	Open and Honest Communication
	Proactiveness
	Direct Customer Involvement
	Collective Ownership
	Blame Sharing
	Negotiation
	Conflict Resolution
Masculinity (MAS)	Authoritative
	Empowered
	Negotiation
	Conflict Resolution
Uncertainty Avoidance Index (UAI)	Risk taking
	Innovation
	Tolerance for Change
	Meeting deadlines and expectations
	Time keeping
	Conflict Resolution
Long term Orientation (LTO)	Trust people more than process
	Risk taking
	Quick Decision Making
	Meeting deadlines and expectations
	Time keeping

Table 6. Previous research in similar fields

Previous Methodology Studies	**Previous Societal Culture Studies**
Actual use of methods in UK (Fitzgerald (1997), Hardy, Thompson and Edwards (1995), and Chatzoglou (1997) Use of software systems development methods in Brunei (Rahim, Seyal and Rahman (1998) Use of methodologies and CASE tools in Norway (Krogstie (1995)) Adoption of SSADM in a government agency in Australia (Sauer and Lau (1997)	Differences in motivation of analysts and programmers in Singapore and the U.S (Couger (1986) The Effects of Culture on Performance Achieved through the use of Human Computer Interaction (Ford & Gelderblom (2003) Evaluating the impact of Cultural Differences among software Programmers in India and in the US (Maudgalya, 2004) Exploring the Relationships between Individualism and Collectivism and Attitudes towards Counseling among Ethnic Chinese, Australian and American University students (Snider, 2003) Cross-cultural leadership (Grisham, 2006) Culture and International Usability Testing: The effects of Culture in Interviews (Vatrapu, 2002) Cultural Influences and Differences in Software Process Improvement Programs (Wong & Hasan, 2008) An investigative study into the adoption of cross-cultural management practices in selected public and private sector organizations – A grounded theory approach (Reyes, T M, 2005) The Reflexivity between ICTs and Business Culture: Applying Hofstede's theory to compare Norway and the united States (Sornes, Stephens, Saetre & Browning, 2004)



	PDI Power Distance Index	IND Individualism	MAS Masculinity	UAI Unceartainty Avoidance Index	LTO Long Term Orientation
Australia	36	90	61	51	31
India	77	48	56	40	61
United Kingdom	30	89	62	30	20
Japan	50	42	90	88	75

Figure 1. Hofstede's cultural dimensions – Australia, India, United Kingdom, and Japan