

The Impact of National Culture on Software Engineering Practices

Dr Aileen Cater-Steel

Senior Lecturer

Aileen.Cater-Steel@usq.edu.au

Professor Mark Toleman

Professor of Information Systems

Mark.Toleman@usq.edu.au

School of Information Systems, Faculty of Business

University of Southern Queensland

Toowoomba 4350

Australia

Fax: +61 7 46315594

Abstract

Trends that have contributed to the globalisation of the software engineering industry include virtual collaborative teams, off-shore outsourcing, and international migration of IT professionals. These three trends and the international spread of software engineering standards and methodologies are explored with specific examples from the Australian software engineering industry. Results from a Europe/Australia study about adoption of software best practice conducted in 16 countries are then summarised and analysed using Hofstede's cultural dimensions. The discussion considers the efficacy of the concept 'national culture' in light of the analysis and concludes that software engineering researchers need to reconsider the concept and measurement of national culture. Implications of the globalisation of software engineering standards and methodologies on various stakeholders are discussed.

Keywords

National culture, Hofstede, globalisation, software best practice, software engineering process, global software development standards, virtual teams, off-shore outsourced development.

Biographical notes:

Dr Aileen Cater-Steel is a Senior Lecturer in Information Systems at the University of Southern Queensland (USQ) Australia. Her current research interests are IT service management and software process improvement. She has also published research related to IT governance, software engineering standards, organisational and national culture, and electronic commerce. Prior to her university appointment, Aileen worked in private and government organizations where her career progressed from Programmer, Systems Analyst and Project Manager to IT Manager.

Professor Mark Toleman is Professor and Head of School of Information Systems at the University of Southern Queensland. His research interests are wide and include IT service management, IT governance, systems development methodologies, research-practitioner nexus, novice developers and information systems education. He has published over 100 articles in books, refereed journals and refereed conference proceedings.

1. Introduction

Dramatic improvements in software engineering tools and methods have allowed geographically and culturally diverse developers to collaborate in global software development teams (Karolak 1998). Software development has become a global activity and it is recognised that the business environment and culture varies from one location to another (Shore & Venkatachalam 1995). Added to this, recent migration trends have resulted in a multicultural information communication and technology (ICT) workforce in Australia, and in other countries such as the USA, Great Britain and Ireland. The concept of *national culture* as defined by Hofstede refers to the collective mental programming shared by people which distinguishes the members of one nation from those of other nations (1980). To date, there has been limited research into the role of national culture in software development, and doubts raised about whether national culture, as defined by Hofstede, ever actually existed (Myers & Tan 2002).

This paper explores the relevance of national culture solely in relation to software engineering teams. Other software engineering research has considered the role of national culture in relation to developing systems for a culturally diverse range of users, and the deployment, use and management of international information systems (Myers & Tan 2002, Corbitt et al. 2004, Hazzan & Dubinsky 2005, Thanasankit 2002, Srite 2006). Essentially, the study attempts to validate Hofstede's national culture dimensions for the case of software developers.

In the next section (§2), the emergence of a multicultural software development industry is discussed and the Australian computing professional workforce is analysed to determine the extent to which recent immigration trends have impacted on the local industry. Global trends in software development standards and methodologies are discussed in §3.

In §4, the results from a software development best practice survey carried out by the European Software Institute (ESI), and replicated in Australia are used to highlight variations in the adoption of software practices across 16 countries. Hofstede's (1980) five generic factors which characterise value systems in different national culture dimensions are described and used to explore the relationship between the Europe/Australia study results and the cultural dimensions. The discussion (in §5) focuses on the outcomes of the immigration analysis and survey data, in particular highlighting limitations in the concept of national culture and its shortcomings in explaining issues in software engineering. Implications for stakeholders impacted by globalisation of software engineering standards and methodologies are explored. The conclusion (§6) suggests directions for future research.

2. Globalisation of Software Industry

Over the last decade global development efforts have become the industry norm rather than the exception (MacGregor et al. 2005). Previously, systems were either developed locally, or software development was carried out in countries with relatively mature software industries. With the recent liberalisation of markets and economic progress in many developing nations, emerging countries such as India are increasing in software development capability, and gaining a greater share of the international market (Costlow 2003).

Three recent trends have necessitated the consideration of the effect of national culture in regards to software development practices and teams: geographically dispersed virtual collaborative teams; outsourcing to off-shore companies; and migration of software developers.

2.1 Virtual Collaborative Teams

For multinational organisations, information and communication technologies such as intranets, e-mail, and videoconferencing facilitate the software development work of geographically distributed teams. Organisations are able to reduce travel expenses as well as labour costs by employing staff in countries where labour rates are lower. Improved time to market can be achieved with the 'follow the sun' approach enabling teams of software developers to work on projects 24 hours per day (Edwards & Sridhar 2003).

The use of widely dispersed collaborative teams is prompting a radical change in the way software is developed (Karolak 1998). Promoted as a just-in-time approach, organisations such as Bangalore-based Infosys provide low-cost, world-wide application developers and use Internet-based, open-source tools to create application development teams in collaboration with their clients' IT staff (McCarthy 2003). Such teams are able to respond quickly when new applications are required urgently, in contrast to in-house development projects which are often delayed due to the organisation's system development backlog.

When a team is composed of staff from a variety of nationalities, with little or no face-to-face contact, cultural issues are inevitable. Telecommunication technologies may impact cultural differences, in some cases ameliorating them, and in others exacerbating them (Olson & Olson 2003).

2.2 Off-shore Outsourcing

In recent years, there has been a marked increase in off-shore software development. In Australia for example, Loane (2003) reports that only 25 per cent of the A\$4 billion software sold in Australia each year is developed by Australian firms. As more organisations outsource IT development to off-shore firms, thousands of developer jobs have been lost from the Australian IT industry (Foresheew 2004). With this increasing trend of large organisations outsourcing their software development to firms in countries such as India, the local software industries are beginning to realise the importance of adopting internationally recognised practices to improve quality and productivity.

Off-shore outsourcing may also include virtual collaborative teams with developers from multiple locations working together. Edwards and Sridhar (2005) studied 24 virtual teams in Canada and India finding that trust and a well-defined task structure were important positive factors for global virtual teams.

2.3 International Migration of IT Professionals

In response to a skills shortage, Australia's immigration policies were changed in 2001 to encourage ICT professionals to migrate to Australia. This policy has been revised several times since 2001 as it became apparent that the skills shortfall had been grossly overestimated and migrants were competing with unemployed locals for a shrinking number of IT positions

(AAP 2003). To balance the supply and demand of IT professional migrants, the current *Migrants on Demand List*, which came into effect in September 2006, is specific in stating nine specialisations for computing professionals: CISSP (certified information systems security professionals); C++/C#/C; Java; J2EE; Network Security; Oracle; PeopleSoft; SAP; and Siebel (DIAC 2006).

Another source of migrant IT workers is international student graduates from Australian universities. For example, in 2006 a total of 172,297 international students were studying at Australian Universities. Of this number, 21,000 were enrolled in Computer Science and Information Systems courses and many of the 74,000 students in business administration and management would also be undertaking IS/IT specialisations (AEI 2007).

As a result of the approval of permanent and temporary visas, the Australian IT industry now has many computer professionals from a wide range of birthplaces. An analysis of the 2001 Australian Census revealed that of the 183,000 IT workers, 40 percent of computing professionals, 35 percent of IT Managers and 32 percent of Computing Support Technicians were not born in Australia (ABS 2005). The majority of permanent and temporary migrant computing professionals recorded their birthplace as Asia (41%) or Europe (35%). Considering the relative populations of countries, the proportion of migrants from New Zealand (6%) is very high, probably due to its close proximity, common language and similar culture to Australia. Of the remainder, the breakdown was Africa (6%), North America (5%) and other (7%). Since the 2001 census, the proportion of migrant IT workers has increased further with 18,000 Computing Professionals granted 'skill stream' visas in the two years from July 2003 to June 2005 (Vanstone 2005). Consequently, the multicultural diversity of software development teams in Australia has increased substantially.

3. Global Trend Towards Software Development Standards

In order to maintain a role in the domestic and international market, software firms are under pressure to comply with recognised software process improvement programs and undertake third party assessments to provide evidence to investors and customers of the firm's commitment to software quality (Saran 2001). The most widely adopted framework, the Capability Maturity Model Integrated (CMMI) was developed by the Software Engineering Institute (SEI) at Carnegie Mellon University in Pittsburgh. Evidence of the CMMI's international acceptance by the software development community is apparent by the report that only 36 percent of the 1,581 CMMI appraisals performed up to mid 2006 involved USA-based organisations (SEI 2006). Of these, 23 appraisals were conducted in Australia (SEI 2006).

Local Australian software firms have been encouraged to adopt global standards in order to be competitive. For example, the Australian Computer Society (ACS) issued a policy statement on software quality accreditation stating that 'all organisations involved in software engineering should implement sound, auditable, management and process improvement principles by adopting an effective third party quality assurance process such as ISO 9001:2000; ISO 15504, CMMI, or other appropriate standard according to the firm's needs and requirements' (ACS 2004 p.5). Increasingly, large Australian software purchasers, such as the Defence Materiel Organisation (DMO), Telstra and ANZ Bank, are recognising CMMI benchmark results when selecting their suppliers (Howarth 2004).

Consequently, local development firms face challenges of adopting and adapting systems development methods created in other countries and cultures (Shore & Venkatachalam 1995, Krutchen 2004). In commenting on the fact that the CMMI and ISO 15504 have two dimensions, the process dimension and capability dimension, Biro et al. call for a third dimension to CMMI and ISO 15504—the cultural dimension—because ‘the national cultural position of the company may determine a different meaning and suitable improvement actions’ (2002 p.36). Table 1 provides a summary of the cultural maturity levels proposed by Biro et al.

Table 1. Summary of Cultural maturity levels (Source: derived from Biro et al. 2002)

<i>Cultural Maturity Level</i>	<i>Description of Process Area at Specified Level</i>
0 Closed	The specific and/or generic practices are prescriptive to the extent where no differences in cultural value systems are allowed.
1 Open	The specific and/or generic practices are open enough to allow for differences in cultural value systems.
2 Model based	The consideration of cultural differences is based on a scientifically established model. Hofstede’s multidimensional model is an example, but other models are also acceptable in case they are useful in distinguishing the differences in cultural value systems that have an impact on the performance of the specific and/or generic practices.
3 Comprehensive	The scientifically established cultural model is comprehensively applied to all specific and generic practices leading to the achievement of the specific and/or generic goals of the process area at the given capability level.
4 Tailored	The depth and complexity of the application of the cultural model to the specific and generic practices is based on quantitatively managed experience and business needs.
5 Competency driven	The cultural model applied to the specific and generic practices is refined, extended, or fully changed on the basis of competency acquired through quantitatively managed long-term model experience and business needs.

National culture was found to have an important impact in the implementation of software development methodologies such as extreme programming in Israel (Hazzan & Dubinsky 2005) and soft systems methodology in Hong Kong (Moores & Gregory 2000).

For large multinational corporations, it is often difficult to balance their methods and systems to accommodate the local needs of host organisations, as well as the centralised needs of headquarters (Cheung & Burn 1994). Furthermore, critical success factors for IT vary

depending on geographic region, therefore, it is important for multinational corporations to understand the different sets of issues so that management can take appropriate actions to achieve success (Khandelwal & Ferguson 1999).

Despite the recent trend towards standard practices and methodologies, there is apparently still a wide variation of practices in use depending on the size of the project and the type of project (for example, MIS, military, outsourced, commercial) (Jones 2003). This variety in software development practice has implications for organisations purchasing software, contracting for off-shore outsourcing, and managing collaborative teams with diversity in terms of geographic location, or co-located multicultural team members.

4. Software Development Practices – Lack of International Homogeneity

Although many authors refer to software developers using dominant, prevalent, or common practices, research to date has returned conflicting reports regarding actual current use. After examining more than 12,000 projects, Jones (2003) concluded that no single development method is universally deployed and that there is little uniformity in activities relating to requirements gathering, design, coding, and defect removal. He noted ‘over 40 methods for gathering requirements, over 50 variations in handling software design, over 700 programming languages, and over 30 forms of testing’ (p.25). This section reports on the results of studies comparing the variation in adoption of software development practices in various countries.

4.1 International Comparative Studies

Throughout the 1990s, a number of studies aimed to determine if significant differences existed in software development practices. The outcome of this research effort did not produce a consensus view, apart from the call for further research. Recently, Cusumano et al. (2003) compared a total of 104 projects from India, Japan, Europe and the US from the point of view of improved practices in relation to outsourcing. Their study suggests that India’s adoption of CMMI has resulted in higher quality practices there.

Despite this considerable research effort, there is little consensus that the underlying factors responsible for the similarities and differences in software development practices are based on the geographic location or nationalities of the developers.

4.2 Europe/Australia Study

The most widely reported survey of best practice in Europe was that conducted by the European Software Institute from 1995 to 1997 (Dutta et al. 1998b, ESI 1996). Organisations were encouraged to apply for funding to enable them to adopt a specific software process improvement (SPI) project in a real-life commercial environment. The Software Best Practice Questionnaire (SBPQ) was distributed as part of the call for proposals for funding. A best practice is defined as ‘a management practice that is widely recognised as excellent and is recommended by most practitioners and experts in the field’ (ESI 1997). The SBPQ represents the ‘subjective consensual views of multiple experts’ (Dutta & Van Wassenhove 1997), and comprises a subset of core software development practices including

organisational issues, standards and processes, metrics, control of the development process, and tools and technology.

The ESI study yielded valuable findings from the analysis of the 1,279 responses received over three years. The third and final survey in 1997 generated 397 responses (ESI 1997) and showed 'wide variation in both awareness and application of process improvement techniques' (Dutta et al. 1999). The ESI survey was replicated as a mail survey in Queensland (Australia) in 1998 (Cater-Steel et al. 2005). The survey returned 205 responses from organisations which develop software for sale or for internal use.

Table 2 shows the average adoption by country for the Europe/Australia SBPQ study. The difference in the average adoption varied markedly: firms from the United Kingdom and France showed the highest overall adoption rates, while Sweden and Spain had adopted the lowest proportion of practices. For specific practices, adoption varied greatly from one country to another. For example, while 83 percent of Norwegian firms establish a change control function for each project, only 20 percent of Belgian organisations have adopted this practice. Both Belgium and Denmark scored an average of 53 percent for maintaining awareness of CASE or other new software development technologies compared to only 8 percent of Irish firms. While Spain scored poorly on most practices, it was the leader for controlling estimates, schedules and changes, and also obtaining signoff from all parties before changing plans (Dutta et al. 1999). Australian organisations showed strengths in the use of standards and processes (54% adoption), but were very weak in the application of metrics (35% adoption).

Table 2. Best Practice average adoption for each country– overall and for each group of processes (Source: ESI 1997)

<i>Country</i>	<i>N</i>	<i>Org. Issues</i>	<i>Standards & Processes</i>	<i>Metrics</i>	<i>Control</i>	<i>Tools</i>	<i>Overall Average Adoption</i>
France	18	72	62	61	76	58	65%
U.K.	52	66	63	52	67	50	60%
Greece	18	63	57	49	65	50	57%
Denmark	17	64	53	46	63	53	55%
Finland	4	63	56	50	54	50	55%
Austria	16	66	50	42	60	46	53%
Norway	6	60	53	44	61	48	53%
Italy	77	57	52	50	61	40	52%
Germany	62	55	48	43	52	47	49%
Netherlands	30	57	49	41	51	48	49%
Australia	205	48	54	35	49	47	48%
Israel	11	57	47	38	55	34	46%
Ireland	12	51	43	36	51	45	45%
Spain	34	53	44	36	57	35	44%
Belgium	15	52	41	40	46	40	43%
Sweden	13	38	36	25	33	26	32%

(Note ESI results are from 1997 survey. Countries with less than 4 responses were omitted.)

As world class standards are dynamic, the set of practices considered to be the best changes over time, especially in software development which has frequently adapted to changes brought about by evolution of technology (Finkelstein & Kramer 2000). With the passing of time, best practice becomes standard practice as other superior practices emerge (Cragg 2002). The SBPQ heavily emphasises project management, but has no practices relating to risk management, measurement, validation, joint review or audit. Therefore, it is recognised that the items from the ESI questionnaire may not provide an entirely valid measurement of best practice across the industry. This point is acknowledged by the ESI: 'progress in software engineering may not be visible along dimensions measured in the survey' (ESI 1998, p. 29). Further, challenges in replicating the ESI survey in Australia have been previously reported (Cater-Steel et al. 2005).

The results of the ESI survey highlight disparities across a range of 16 countries in terms of their adoption rates of a set of software development best practices. The next section explores the potential role of national culture in explaining such differences.

4.3 National Culture

In the research and practice arenas of national culture, Geerte Hofstede is considered to be one of the most influential theorists (Hardin et al. 2007). Hofstede's initial research was based on the analysis of 116,000 IBM employees from more than 50 countries surveyed between 1967 and 1973. Hofstede asserts that although people may have similar occupational and organisational culture, national culture, the collective mental programming shared by people which distinguishes the members of one nation from that of other nations, is about deeply held values and is part of the mental software acquired from family and school during the first ten years of life (Hofstede & Hofstede 2005, Hofstede 1980). Hofstede's results have been applied by researchers and verified by many replications (Hofstede & Hofstede 2005). In particular, Myers and Tan (2002) found that 24 of 36 information systems research studies of national culture used Hofstede's theory. For example in IT related research, Frank et al. (2001) found evidence that innovativeness correlates with low uncertainty avoidance in a study of the adoption of mobile technology across Finland, Germany and Greece; and more recently, Borchers (2003) applied Hofstede's theory to understand project problems experienced by project teams of Indian, Japanese and American software developers.

Hofstede characterises national culture using five dimensions as summarised in table 3. Each dimension is 'an aspect of culture which can be measured relative to other cultures' (Hofstede & Hofstede 2005, p.23).

Table 3. Hofstede's National Culture Dimensions (Mahoney et al. 2001)

<i>Dimension</i>	<i>Low Score Value</i>	<i>High Score Value</i>
Power distance	society de-emphasizes the differences between citizen's power and wealth	inequalities of power and wealth within society
Individualism vs collectivism	collectivist nature with close ties between individuals	individualism and individual rights are paramount
Uncertainty avoidance	tolerance for variety of opinions, less concern about ambiguity and uncertainty	low tolerance for uncertainty and ambiguity
Masculinity vs femininity	value social relevance, quality of life, welfare of others	aggressive goal behaviour, high gender differentiation, males dominate
Long term vs short-term orientation	place less emphasis on hard work, perseverance	embraces long-term devotion to traditional, forward thinking values

In relating Hofstede's dimensions to the adoption of best practice techniques, it could be expected that higher adoption may be associated with low uncertainty avoidance (willingness to adopt new techniques), and low individualism (conformance to group working practices). For example, Hofstede's scores indicate that Australians, compared to others, have low uncertainty avoidance (would be quick to adopt innovations) but high individualism (resistant to standard work practices) (Mahoney et al. 2001).

The difference in the ESI best practice adoption levels across Europe raises the question of national cultural influences. Anecdotes characterising differences based on nationality claim that the French are better at object-oriented development; the Japanese excel at metrics; British developers use the Jackson methodology; Belgians are more process-oriented; and US developers code first and design later (Carmel 1999). Using the ESI survey Europe data, Dutta, Lee and Van Wassenhove (1998a) briefly explored the concept of national culture using Ronen and Shenkar's (1985) national culture clusters to compare adoption of clustered countries. Dutta, Lee and Van Wassenhove (1998a) observed that Germany and Austria behaved similarly; however, with respect to Scandinavian countries, they found considerable variance warranting further research. The clusters derived by Ronen and Shenkar measure work goals, values, needs, and job attitudes and are named Anglo, Germanic, Nordic, Latin European, Latin American, with Australia classed in the Anglo cluster along with United Kingdom, Ireland, USA, Canada, New Zealand and South Africa (Mahoney et al. 2001).

Correlation tests were carried out on the data from the Europe/Australia surveys to explore if Hofstede's dimensions scores were associated with adoption rates of software best practices by country. The test results, reported in detail in Cater-Steel and Toleman (2006), indicate no significant correlations of best practice adoption with any of Hofstede's dimensions. Therefore, Hofstede's theory is not validated for the case of software development

organisations across the 16 countries surveyed in the Europe/Australia study. The next section discusses reasons for the lack of correlations, the sample upon which Hofstede based his theory, and also the very notion of national culture.

5. Discussion

As well as critically evaluating Hofstede's dimensions of national culture, this section considers the implications for stakeholders such as governments, professional bodies, managers, and universities and training providers.

5.1 Validity of Hofstede's concept of national culture

Previous research has raised doubts about whether it is reasonable to expect that software development best practice adoption would be related to Hofstede's national culture dimensions. In recent years, Hofstede's analysis and model have drawn criticism from researchers (McSweeney 2002). One of the issues raised by Myers and Tan (2002) and other researchers concerns the ability to generalise Hofstede's scores, considering the limited demographic variation in the population surveyed: the survey data was mainly from male employees of one multinational organisation (IBM) and severely limited in terms of the range of ages of respondents.

Another related issue, also explored by Myers and Tan (2002), is whether national culture remains static—as claimed by Hofstede—or contested, temporal and emergent as claimed by Kahn (1989). In defending the strength of the underlying theory of static national culture in the face of global use of email and other technology, Hofstede and Hofstede believe 'the software of the machines may be globalised, but the software of the minds that use them is not' (2005, p.330). The diversity of birthplaces in the Australian workforce of computing professionals (presented in §2) provides support for the temporal and emerging nature of national culture, influenced by changes in the ethnic and racial mix of the population. It is suggested, therefore, that Hofstede's sample of IBM male employees circa 1970 may not represent the diverse workforce which exists in each country today, and which is gradually changing as more IT professionals undertake international migration.

Therefore, the disparate rates of adoption of software practices by country found in analysing the Europe/Australia survey results may be caused by factors other than the deeply held national cultural values. For example, the practices used by firms may originate from the methods and techniques taught in the curriculum of local colleges and universities, or as a result of individual government purchasing policies promoting various methodologies (such as CMMI). Factors such as these may foster standardisation within the local industry, but may be the source of variation when comparing diverse geographical groups of software development firms. The Europe/Australia best practice survey, although providing a valuable snapshot of the state of practice in many countries, was designed to measure behaviour, not culture. Recently, there has been growing awareness of the importance and challenges of researching cross-cultural research (Straub et al. 2002, Corbitt et al. 2004, Karahanna et al. 2002). Dimensions of national culture are very difficult to conceptualise and measure. It has been recommended that in-depth case studies, discourse analysis and ethnographies are required rather than surveys (Myers & Tan 2002, Sharp et al. 2000). In Walsham's (2002) study of software production in teams composed of Jamaican and Indian developers, use of

structurational theory and analysis was considered superior to Hofstede's model. Walsham considered that Hofstede-type studies could be used to describe contradictions between different cultures but do not enable the analysis of how such contradictions could result in conflict.

Considering the growing criticism of Hofstede's model, the question raised is *why do researchers continue to use it?* It is certainly convenient for the purpose of statistical analysis to have a complex concept of national culture reduced to a 'simplistic categorical description' of five dimensions each with a numerical score for each country (Corbitt et al. 2004). A review of journals in the computing disciplines from 1995-1999 showed that quantitative studies tend to dominate over qualitative studies (Glass et al. 2004), indicating a possible bias on the part of researchers or journal editors.

5.2 Globalisation of Software Development - Implications for Stakeholders

The key stakeholders affected by the globalisation of software development are the project managers and software developers and ultimately their clients. In this section, the implications of globalisation of software development on the following stakeholders is discussed: government agencies, professional bodies, project managers, and universities and training providers.

Government policy makers need to be aware that immigration policies can have dramatic impacts on the local software development industry. As illustrated in section 2.3 with the Australian example, it is vitally important to have accurate estimates of specific skill shortages and flexibility to balance the demand for workers with immigration policies. Governments also play an important role in nurturing the local IT industry by implementing policies and providing funds for software process improvement programs. In Singapore, Palvia and Hunter (1996) found proactive planning and strategizing by the government significantly increased the rate of transfer of state-of-the-art IS development methods from more advanced countries. Government policies to outsource software development to off-shore firms is seen to damage the local IT industry.

Professional Bodies (such as the Australian Computer Society) also play an important role in globalisation of the local industry. In Australia, the ACS lobbies the Australian Government in relation to immigration policy and is responsible for accrediting migrants for ACS membership. As well, as described in §3, the ACS encourages its members to adopt international standards. The ACS also influences the computing curricula of universities and training providers by accrediting courses for graduate membership.

Project managers responsible for collaborative teams and off-shore development projects need to be aware of cultural differences. With the increase in immigration, and growing prevalence of virtual multinational teams, software project managers need to make extra efforts to tackle cross-cultural issues and create an environment that takes into account the cultural norms of the staff involved (Burn et al. 1995, Krishna et al. 2004).

As well as providing training to all staff to assist them understand cultural differences, managers need to understand cultural differences in relation to communication, authority, hierarchy, relationships, and contractual agreements. For example, research has found that software engineering teams in US, India, China and Hungary have different ways of

coordinating work, variations in flexibility of work hours, intra-group communication mechanisms and contrasting ideas about helping each other (Perlow & Weeks 2002, Perlow 2001). Other researchers have established marked national cultural variations in attitudes to meeting deadlines (Walsham 2002) and in communicating bad news about the progress of IT projects (Tan et al. 2003). Understanding the influences of national culture will assist multicultural IS organisations to select, adapt, and better manage process methodologies (Thanasankit 2002). As well as training, managers should be encouraged to provide extensive documentation since software development practices, as processes, may vary from one country to another.

Universities and other training providers should be aware of global standards and methodologies to ensure graduates are prepared for the international software engineering workplace (Cater-Steel & Toleman 2007). It is the responsibility of institutions to be aware of industry demand for specific skills to assure graduates of employment. Another important role is to educate students to value diversity based on race and ethnicity. This would include understanding of various cultures and tolerance for the ideas of others from different backgrounds. Such a program in multicultural understanding would have numerous benefits to students, academics, universities and industry.

6. Conclusion

The software engineering industry has become globalised due to trends such as the maturation of software industries in developing countries, virtual collaborative teams covering extended geographic areas, off-shore outsourcing and international migration of computing professionals. Multinational corporations, software purchasers, firms undertaking off-shore outsourcing, and firms with teams of local or distributed developers all need to be aware that practices used by software developers vary according to geographic location, in the same way senior managers need to be aware of local business practices in negotiating contracts with international business partners.

The migration analysis provided a profile of the multinational nature of the Australian computing workforce. This workforce analysis provided evidence for the argument to view national culture as temporal and emergent, rather than the notion that national culture is static. Furthermore, the analysis of the Europe/Australia best practice study failed to prove a link between Hofstede's cultural dimensions and practices surveyed in 16 countries. Consequently, researchers are advised against the use of simplistic frameworks such as that espoused by Hofstede, and are encouraged to explore the concept of national culture with appropriate research methodologies.

Currently, there is little published information on international migration of IT workers and performance of multicultural teams. Furthermore, to date there has been little software engineering research on the impact of government policies, and the professional bodies which influence government agencies. There is a wide range of policies involved including immigration, industry development, and procurement. The issues discussed are also of vital importance to project managers and education providers who are urged to provide training to ensure multicultural collaborative teams of software engineers are productive and motivated to work harmoniously together.

Acknowledgement

The authors thank the ESI for granting permission to use the Best Practice Questionnaire, and the Software Quality Institute of Griffith University for funding the survey. Associate Professor Terry Rout of Griffith University provided valuable assistance during the design and execution of the study.

References

- AAP (2003) ICT jobs cut from govt's migrant skills list, URL www.smh.com.au/articles/2003/11/12/1068329608544.html, Accessed 9 July.
- ABS (2005) 2001 Census of Population and Housing: Number of Employed Persons by State of Usual Residence by Birthplace by Occupation. *Unpublished*. Brisbane.
- ACS (2004) Policy Statement on Software Quality Accreditation. Sydney, Australian Computer Society.
- AEI (2007) International Student Data for 2006. Canberra, Australian Education International, Australian Department of Education, Science and Training, Australian Government.
- Biro, M., Messnarz, R. & Davison, A. (2002) The impact of national cultural factors on the effectiveness of process improvement methods: the third dimension. *Software Quality Professional*, 4:4, 34-41.
- Borchers, G. (2003) The software engineering impacts of cultural factors on multi-cultural software development teams. *Proceedings of the 25th International Conference on Software Engineering*. Portland, Oregon, IEEE Computer Society Washington, DC, USA.
- Burn, J. M., Ng Tye, E. M. W. & Ma, L. C. K. (1995) Paradigm Shift - Cultural Implications for Development of IS Professionals. *Journal of Global Information Management*, 3:2, 18-28.
- Carmel, E. (1999) *Global software teams: collaborating across borders and time zones*, Upper Saddle River, NJ, Prentice Hall.
- Cater-Steel, A. P. & Toleman, M. (2007) Education for IT Service Management. *International Journal of IT Standards and Standardization Research*, 5:2, 27-42.
- Cater-Steel, A. P. & Toleman, M. A. (2006) Exploring National Cultural Issues in Software Development Practices. *EuroSPI*. Joensuu, Finland, LCNS.
- Cater-Steel, A. P., Toleman, M. A. & Rout, T. P. (2005) Addressing the Challenges of Replications of Surveys in Software Engineering Research. *International Symposium on Empirical Software Engineering*. Noosa.
- Cheung, H. K. & Burn, J. M. (1994) Distributing global information systems resources in multinational companies: a contingency model. *Journal of Global Information Management*, 2:3, 14-27.
- Corbitt, B. J., Peszynski, K. J., Inthanond, S., Hill, B. & Thanasankit, T. (2004) Cultural Differences, Information and Code Systems. *Journal of Global Information Management*, 12:3, 65-85.
- Costlow, T. (2003) Globalization drives changes in software careers. *IEEE Software*, 20:6, 14-16.
- Cragg, P. B. (2002) Benchmarking information technology practices in small firms. *European Journal of Information Systems*, 11:4, 267-282.
- Cusumano, M. A., MacCormack, A., Kemerer, C. F. & Crandall, B. (2003) Software development worldwide: the state of the practice. *IEEE Software*, 20:6, 28-34.
- DIAC (2006) Visas & Immigration: professionals and other skilled migrants, URL <http://www.immi.gov.au/skilled/general-skilled-migration/skilled-occupations-in-demand.htm>, Accessed 25 Feb.
- Dutta, S., Lee, M. & Van Wassenhove, L. (1999) Software engineering in Europe: a study of best practices. *IEEE Software*, 16:3, 82-90.
- Dutta, S., Lee, M. A. & Van Wassenhove, L. N. (1998a) Adoption levels of software management best practices in European organizations. *European Conference on Software Process Improvement*. Monte Carlo.
- Dutta, S. & Van Wassenhove, L. N. (1997) An empirical study of adoption levels of software management practices within European firms. Fontainebleau, France, INSEAD.
- Dutta, S., Van Wassenhove, L. N. & Kulandaiswamy, S. (1998b) Benchmarking European software management practices. *Communications of the ACM*, 41:6, 77-86.
- Edwards, H. K. & Sridhar, V. (2003) Analysis of the Effectiveness of Global Virtual Teams in Software Engineering Projects. *36th Hawaii International Conference on System Sciences*. IEEE.
- Edwards, H. K. & Sridhar, V. (2005) Analysis of Software Requirements Engineering Exercises in a Global Virtual Team Setup. *Journal of Global Information Management*, 13:2, 21-41.

- ESI (1996) Software Best Practice Questionnaire Comparative Analysis of 1995 and 1996. Bilbao, European Software Institute.
- ESI (1997) 1997 Software Best Practice Questionnaire: Analysis of Results. Bilbao, European Software Institute.
- ESI (1998) Software Best Practice Questionnaire: Trend Analysis 1995-1997. Bilbao, European Software Institute.
- Finkelstein, A. & Kramer, J. (2000) Software engineering: a roadmap. IN Finkelstein, A. (Ed.) *The Future of Software Engineering*. ACM Press.
- Foreshew, J. (2004) Developer offshoring to increase. *The Australian*.
- Frank, L., Sundqvist, S., Puumalainen, K. & Taalikka, S. (2001) Cross-cultural comparison of innovators: empirical evidence from wireless services in Finland, Germany and Greece. *ANZMAC Conference*.
- Glass, R. L., Ramesh, V. & Vessey, I. (2004) An Analysis of Research in Computing Disciplines. *Communications of the ACM*, 47:6, 89-94.
- Hardin, A. M., Fuller, M. A. & Davison, R. M. (2007) I Know I Can, But Can We? Culture and Efficacy Beliefs in Global Virtual Teams. *Small Group Research*, 38:1, 130-155.
- Hazzan, O. & Dubinsky, Y. (2005) Clashes between culture and software development methods: the case of the Israeli hi-tech industry and extreme programming. *Agile Conference*. Denver Colorado, IEEE.
- Hofstede, G. (1980) Motivation, leadership, and organization: do American theories apply abroad? *Organizational Dynamics*, Summer, 42-63.
- Hofstede, G. & Hofstede, G. J. (2005) *Cultures and Organizations: Software of the Mind*, New York, McGraw-Hill.
- Howarth, B. (2004) Software shut-out. *Business Review Weekly*, 26:1, 29-31.
- Jones, C. (2003) Variations in software development practices. *IEEE Software*, 20:6, 22-27.
- Kahn, J. S. (1989) Culture: demise or resurrection? *Critique of Anthropology*, 15:1, 32-49.
- Karahanna, E., Evaristo, J. R. & Srite, M. (2002) Methodological Issues in MIS Cross-Cultural Research. *Journal of Global Information Management*, 10:1, 48-55.
- Karolak, D. W. (1998) *Global Software Development: managing virtual teams and environments*, Los Alamitos, CA, IEEE Computer Society Press.
- Khandelwal, V. K. & Ferguson, J. R. (1999) Critical success factors (CSFs) and the growth of IT in selected geographic regions. *32nd Hawaii International Conference on System Science*.
- Krishna, S., Sahay, S. & Walsham, G. (2004) Managing cross-cultural issues in global software outsourcing. *Communications of the ACM*, 47:4, 62-66.
- Krutchen, P. (2004) Analyzing intercultural factors affecting global software development – a position paper, URL <http://gsd2004.uvic.ca/camera/kruchten.pdf>, Accessed 20 June.
- Loane, S. (2003) Exporting small software firms: a cross-national comparison of the export success factors of software firms. *Software (SEA National)*, 49-52.
- MacGregor, E., Hsieh, Y. & Kruchten, P. (2005) Cultural patterns in software process mishaps: incidents in global projects. *Human and Social Factors of Software Engineering (HSSE)*. St Louis, Missouri.
- Mahoney, D., Trigg, M., Griffin, R. & Pustay, M. (2001) *International Business: A Managerial Perspective*, Pearson Education Australia.
- McCarthy, J. (2003, 18 June) Building Apps on the Fly. *Information Age*, June/July, 49-50.
- McSweeney, B. (2002) Hofstede's model of national cultural differences and their consequences: a triumph of faith - a failure of analysis. *Human Relations*, 55:1, 89-118.
- Moore, T. T. & Gregory, F. H. (2000) Cultural Problems in Applying SSM for IS Development. *Journal of Global Information Management*, 8:1, 14-19.
- Myers, M. D. & Tan, F. B. (2002) Beyond models of national culture in information systems research. In Szwczak, E. & Snodgrass, C. (Eds.) *Human Factors in Information Systems*. Hershey, PA, IRM Press.
- Olson, J. S. & Olson, G. M. (2003) Culture Surprises in Remote Software Development Teams. *Distributed Development*, 1:9.
- Palvia, S. & Hunter, M. G. (1996) Information Systems Development: a Conceptual Model and a Comparison of Methods Used in Singapore, USA and Europe. *Journal of Global Information Management*, 4:3, 5-17.
- Perlow, L. (2001) Time to coordinate: toward an understanding of work-time standards and norms in a multicountry study of software engineers. *Work and Occupations*, 28:1, 91-111.
- Perlow, L. & Weeks, J. (2002) Who's helping whom? Layers of culture and workplace behavior. *Journal of Organizational Behavior*, 23, 345-361.
- Ronen, S. & Shenkar, O. (1985) Clustering countries on attitudinal dimensions: a review and synthesis. *Academy of Management Review*, 10:3, 435-454.

- Saran, C. (2001) The road to software excellence for SMEs, URL www.cw360.com/articles/article.asp?liArticleID=104469&liFlavourID=1, Accessed 6 October 2003.
- SEI (2006) Process Maturity Profile: CMMI v1.1 SCAMPI v1.1 Class A Appraisal Results, 2006 Mid-Year Update, URL <http://www.sei.cmu.edu/appraisal-program/profile/pdf/CMMI/2006sepCMMI.pdf>, Accessed 12 Feb.
- Sharp, H., Robinson, H. & Woodman, M. (2000) Software engineering: community and culture. *IEEE Software*, 17:1, 40-47.
- Shore, B. & Venkatachalam, A. R. (1995) The role of national culture in systems analysis and design. *Journal of Global Information Management*, 3:3, 5-14.
- Srite, M. (2006) Culture as an explanation of technology acceptance differences: an empirical investigation of Chinese and US users. *Australasian Journal of Information Systems*, 14:1, 5-25.
- Straub, D. W., Loch, K., Evaristo, J. R., Karahanna, E. & Srite, M. (2002) Toward a Theory-Based Measurement of Culture. *Journal of Global Information Management*, 10:1, 12-23.
- Tan, B. C. Y., Smith, H. J., Keil, M. & Montealegre, R. (2003) Reporting Bad News About Software Projects: Impact of Organizational Climate and Information Asymmetry in an Individualistic and a Collectivistic Culture. *IEEE Transactions on Engineering Management*, 50:1, 64-77.
- Thanasankit, T. (2002) Requirements engineering—exploring the influence of power and Thai values *European Journal of Information Systems*, 11:2, 128-141.
- Vanstone, A. (2005) Record number of skill stream migrants in 2004-2005. Canberra, Department of Immigration and Multicultural and Indigenous Affairs.
- Walsham, G. (2002) Cross-cultural software production and use: a structural analysis. *MIS Quarterly*, 26:4, 359-380.