GLOBAL SOFTWARE DEVELOPMENT: STRATEGIC IMPLICATIONS FOR U.S. INFORMATION SYSTEMS ACADEMIC PROGRAMS

Balakrishnan Muthuswamy, Independent Consultant
Galen B. Crow, Illinois State University, gbcrow@ilstu.edu

ABSTRACT

Over the past decade the Information Technology (IT) industry has seen an enormous movement towards globalization. In the U.S. there has been an increasing reliance on the use of temporary H-1B visa IT workers, and a rapid growth in offshore outsourcing of IT services. These trends are indicative of a new software development paradigm, Global Software Development (GSD). This paper explores the impact of GSD on the academic preparation of Information Systems graduates.

Keywords: Global Software Development, IS Education, Outsourcing, Off-Shore Software Development.

INTRODUCTION

Over the past decade the Information Technology (IT) industry has witnessed an enormous movement towards globalization. With advanced telecommunications and the emergence of an educated workforce in developing nations, large scale IT initiatives often involve persons from and/or in multiple nations. In the U.S. there has been an increasing reliance on the use of temporary H-1B visa IT workers. The use of H-1B employees has continued to expand even as demand for IT workers has slowed (11). Many of these visa holders work as contract employees providing software development labor. In addition, there is an increasing trend towards the outsourcing of IT work to offshore organizations in developing countries. For example, during 1995-2000, the outsourcing of user support, voice network management, data network management, and data center operations increased dramatically (7). The percentage of Fortune 500 companies who use offshore programmers has doubled over the last three years to around 45% (11).

Thus, U.S. software development environments are becoming global in nature, with team members who have diverse national and ethnic backgrounds. Further, the teams may be geographically distributed across the world. For example, new product development projects at Lucent Technologies now typically involve at least 100 people of varying ethnic backgrounds working across globally distributed locations and organizations (9). This paradigm of software development has been called "Global Software Development" (GSD).

GSD impacts how software development projects are managed in organizations. GSD introduces new factors into the work environment such as global distribution of work groups, time differentials, and cross-cultural differences (2). With GSD each stage of the software development life cycle will entail these additional complexities. For example, during the requirements gathering phase, there is a need for enhanced group modeling tools to support communication, coordination and control of work among the distributive and diverse team members. Email and messaging are useful but do not meet the requirements and complexities of global team distribution (14).

Researchers are only beginning to investigate the phenomenon of GSD. Several issues relevant to GSD have been identified in the literature. Oppenheimer noted GSD-related problems in defining roles and responsibilities regarding information management and organization, relationships and culture, project management, training, requirements engineering, and testing (9). Bowen presents a workflow engine with a defect tracking system for management of GSD projects (1). Maidantchick notes that many common work-related issues are magnified when teams members are geographically disperse (8). Heeks, et. al. define strategies for minimizing gaps in congruence between international clients and subcontractors along six dimensions: coordination/control systems, objectives and values, capabilities, processes, information, and technology (3).

Although some GSD-related research is in progress, more is required. IS educators need to begin to consider this issue as it relates to IS programs, curricula, and courses. IT workers will need education and training on how to operate within the GSD paradigm. This paper identifies some generic skills necessary to manage and develop GSD projects, and provides suggestions on how these skills might fit into U.S. Information Systems (IS) academic programs.

BACKGROUND AND TRENDS IN IT INDUSTRY

Through the latter part of the 20th century the U.S. software industry was the dominant force in world software development. As the 21st century approached, worldwide demand for software grew rapidly. Developed and developing nations became even more dependent upon software. World demand for software and software producers far exceeded supply. As “dot.com” businesses multiplied, there was a burgeoning demand for Internet and web developers. Combined with the Y2K scare of the late 1990’s, shortages in the software development workforce reached all-time highs. As many as 400,000 U.S. information technology jobs were unfilled in 1999 (6).

With the increased world demand and resulting labor shortages, international organizations began building a capacity for software production. India in particular was well situated to create a software development industry. India set high national education priorities for IT graduates and began to rapidly build a surplus of English-speaking IT workers. Software centers were set up in India and other developing countries to develop software for export. These centers could produce software at relatively inexpensive rates by U.S. standards, and also meet emerging software quality standards (10).

After the turn of the century (2000) the demand for IT workers and software developers declined sharply. Y2K was over, many new high-tech businesses failed, terrorism concerns rose, and a general worldwide economic downturn ensued. By early 2002 the U.S. saw a 5% reduction in the IT workforce (6). The prevailing wisdom would argue that a U.S. economic recovery would quickly transform the current IT labor surplus into a shortage again. Indeed, the Department of Labor still predicts software and computer employment to be among the fastest growing occupations in the next decade (13). Other analysts as well predict that despite recent slowdowns, IT employment shortages are inevitable (6).

Globalization is not new to the U.S. work environment, particularly the manufacturing industry. The outsourcing of IT services and software development is beginning to resemble the pattern of the U.S. automobile industry during the latter part of the 20th century. In 1947 the U.S. commanded 82% of world auto production. Cost advantages and quality production in developing countries contributed to shifting of automobile manufacturing facilities to offshore locations. The U.S. share of world production slowly declined to around 26% by 1990 (12). There are many similarities between the current situation in the software development industry and the auto market of 1970-1990. These include the growth of a well-trained global workforce, a rising global demand for products, increased global competition emphasizing cost and quality, a severe economic downturn, and a large drop-offs in U.S. industry employment. To respond to these challenges, the U.S. auto industry initially sought trade protection from international competition. Eventually domestic companies adapted a paradigm of global component production, assembly, and distribution and introduced quality production methods similar to international competitors.

The analogy with the auto industry strengthens the belief that the new GSD paradigm in the IT industry is not a temporary aberration. Although the economic conditions and circumstances under which GSD has developed are cyclical, the trend towards globalization is likely to be permanent. Cost reductions and labor shortages may be primary reasons that industries globalize, but effective and efficient service delivery on a global scale will continue to make GSD a reality. International outsourcing is no longer a remedy for temporary circumstances; it is a long-term corporate strategy (7).

STRATEGIC IMPLICATIONS TO IS EDUCATION

The GSD paradigm impacts the software development environment, and thus IS education, in many ways. Three primary strategic implications to IS education caused by the emerging GSD paradigm are:

- Multi-culturally diverse workforce.
- Global distribution of work.
- Quality software development processes.

Multi-Culturally Diverse Workforce

Work teams in U.S. IT organizations are becoming increasingly multi-cultural. This is due to shifts in domestic demographics as well as to the influx of a large number of H-1B temporary workers from the developing countries. Increasing cultural diversity in the workforce often strains the communication, trust, and collaborative aspects of teams. In addition, offshore outsourcing of IT work has led to the use of diverse teams whose members may also be geographically distributed. Communication, and/or the lack of effective communication, is the most important factor in the success or failure of global software development efforts. It is not uncommon for diverse projects teams to orient themselves around their differing cultural backgrounds rather than their technical skills or role assignments. Even among English speaking teams with multi-cultural backgrounds, communication is complex. Issues arise due to differences in ethnic backgrounds, localized use of words, slang, pronunciation, body language, verbal skills, and variances in work environment procedures and assumptions. Other cultural factors include food habits, values, work ethic, and varied reward structures. Inexperience and misunderstanding of cultural differences often result in inefficient and perhaps dysfunctional work teams. Project teams sometimes splinter into the “we’s” and “they’s” based on cultural background (9).

Educational programs should include culturally integrated classroom experiences, and increased exposure to the diverse economic, social, and cultural practices in the workplace (4). Ideally, becoming familiar with aspects of a different culture would be through classroom education followed a real-world experience of living in that culture. Since this is not always practical, other ways that IS programs might enhance multi-cultural exposure for students include:

- Internships in organizations with a multi-cultural workforce.
- International study abroad.
- International exchange students in U.S. programs.
- U.S. and international students sharing residence halls.
- Foreign language course requirements focusing on developing countries.
- Course projects and assignments with international focus.
- Student project teams composed of multi-culturally diverse members.

Global Distribution of Work

In GSD the workflow and products may be distributed across the world. Communication is often not face-to-face but may be primarily through electronic, digital, or paper means. There is often a time difference between members in distributive teams. India, for example, is about 12 hours ahead of U.S. time. Project planning, work assignment and scheduling, and task coordination is much more complex than in traditional work settings. The management of globally distributed work requires different mindsets, methods and tools.

To prepare students for environments with the additional complexity of global work distribution IS programs should include education and training using tools that enable:

- Synchronous communication for distributed meeting management such as video-conferencing, phone-conferencing, e-chats, instant messaging, e-whiteboard, etc.
- Asynchronous communication for managing “different place, different time” team efforts such as email, common data servers, listserves, message boards, etc.
- Enhanced distributed project management tools that support the software development cycle including distributed work assignments, computer supported collaborative work, distributed document management and control, etc.
- Courses or course topics related to project management issues such as scope control, effort and cost estimation, identifying assignments that are difficult to distribute, procedures/tasks required to manage distributed environment projects, etc.

Quality Software Development Processes

One of the reasons U.S. corporations have been willing to utilize international outsourcing companies for software development is because of the adoption of quality standards by international organizations. Indian producers in particular have adopted quality standards developed in the U.S. but not widely incorporated domestically. The standard gaining most favor internationally is the Capability Maturity Model (CMM), developed by Watts Humphrey from the Software Engineering Institute at Carnegie Mellon University. The top five offshore software providers in India are rated at the highest CMM quality level (10). U.S. organizations have been reluctant to adopt these standards. The U.S. software industry has long been criticized for a lack of quality control, and recently there have been calls by major U.S. corporations to begin taking software quality seriously (5). Managing software quality is further acerbated by the complexities of GSD.

To address the issue of quality standards IS programs should consider including such topics as:

- Software development courses or topics based on the CMM framework.
- CMM-based Personal Software Process (PSP) introduced into programming courses and maintained in all courses where software end products are developed.
- CMM-based Team Software Process (TSP) introduced into course projects involving teams.
- Project management methods such as those certified by the Project Management Institute (PMI).

CONCLUSION

The IT industry has seen a rapid movement towards globalization. The U.S. domestic labor force has become more diverse, including a reliance on H-1B workers. Many U.S. firms are now outsourcing IT development and services to international companies that are known for lower costs and adherence to quality standards. In addition, IT project teams are now more likely to be composed of team members who are geographically and internationally dispersed. This trend has created a new software development paradigm: Global Software Development (GSD). GSD is characterized by the increased reliance on global distribution of software development across multi-culturally diverse work teams. GSD adds to the already complex process of software development. The emergence of the GSD paradigm necessitates software developers to have a deeper understanding and appreciation for multi-cultural workforce issues, to operate in globally distributed work environments, and to incorporate quality standards and processes in the software development life cycle.

U.S. IS educators need to consider these issues in preparing students for the global market. Course curricula need to reflect GSD realities. Students should be provided opportunities to engage in culturally diverse work teams and environments. Students should learn how to develop projects and complete assignments using distributed and asynchronous tools. Students should be taught to incorporate competitive quality standards in all of their development projects across the entire development life cycle. If the U.S. IT and software industries are to maintain a dominant position in the global marketplace, IS educators will need to prepare graduates for the GSD environment.

This paper has presented strategic implications of GSD for IS programs. Further efforts in this area will include specific recommendations for IS programs, accrediting agencies, professional associations, and detailed courses and curricula.

REFERENCES

1. Bowen, S. and Maurer F. Using Peer-to-Peer Technology to Support Global Software Development: Some Intitial Thoughts. *Proceedings of International Workshop on Global Software Development*, Orlando, Florida, May 21, 2002. Web proceedings available at www.cis.ohio-state.edu/~nsridhar/ICES02/GSD/proceedings.html.
2. Carmel, E. *Global Software Teams: Collaborating Across Borders and Time Zones.* Prentice Hall, 1999.
3. Heeks, R., Krishna, S., Nicholoson, B., Sahay, S. Synching or Sinking: Global Outsourcing Relationships. *IEEE Software*, March/April 2001, pp 54-60.
4. Hilton, Margaret. Information Technology Workers in the New Economy. *Monthly Labor Review*, June 2001, pp. 41-45.
5. Hulme, George V. The Sorry State of Software Quality*, InformationWeek*, Jan. 21, 2002, pp. 22-24.
6. Information Technology Association of America (ITAA). *Bouncing Back: Jobs, Skills and the Continuing Demand for IT Workers*, May, 2002.
7. Lackow, Howard M. IT Outsourcing Trends, *The Conference Board Inc.*, ISBN No. 0-8237-0738-5, 2001.
8. Maidantchick, C, and da Rocha, A.R.C. Managing a Worldwide Software Process. *Proceedings of International Workshop on Global Software Development*, Orlando, Florida, May 21, 2002. Web proceedings available at www.cis.ohio-state.edu/~nsridhar/ICES02/GSD/proceedings.html.
9. Oppenheimer, Heather L. Project Management Issues in a Globally Distributed Environment. *Proceedings of International Workshop on Global Software Development*, Orlando, Florida, May 21, 2002. Web proceedings available at www.cis.ohio-tate.edu/~nsridhar/ICES02/GSD/proceedings.html.
10. Overby, Christine Spivey. Offshore Solutions: Skills at a Premium*. The Banker*, January 2002, pp.10-11.
11. Rose, Barbara. High-tech Jobs Flow to Foreign Workers. *Chicago Tribune*, Feb. 10, 2002.
12. Singleton, Christopher J. Auto Industry Jobs in the 1980's: A Decade of Transition. *Monthly Labor Review*, Feb. 1992, 18-27.
13. U.S. Department of Labor, Bureau of Labor Statistics. Occupational Employment Projections to 2010. Published in the November 2001 *Monthly Labor Review*. See: stats.bls.gov/emptab21.htm.
14. Zowgi, Does global Software Development Need a Different Requirements Engineering Process? *Proceedings of International Workshop on Global Software Development*, Orlando, Florida, May 21, 2002. Web proceedings available at www.cis.ohio-tate.edu/~nsridhar/ICES02/GSD/proceedings.html.