

Examining Solutions to Challenges in Offshore Software Development: The Perspective of a SME with a Captive Development Office in India

Jeremy Bellah

Instructor in Operations Management and Information Technology
Rawls College of Business Admin., ISQS Area, P.O. Box 42101, Texas Tech Univ. , Lubbock, TX 79409-2101
(806)742-1547 Fax: (806)742-3193
Email: jeremy.c.bellah@ttu.edu

James R. Burns, Ph.D., P.E., CIRM

Professor of Operations Management and Information Technology
Rawls College of Business Admin., ISQS Area, P.O. Box 42101, Texas Tech Univ. , Lubbock, TX 79409-2101
(806)742-1547 Fax: (806)742-3193
Email: jim.burns@ttu.edu

ABSTRACT

The off-shoring of software development has become popular for US companies during the last 10-15 years. Some US companies outsource entire processes to foreign companies, while others create subsidiaries to perform functions at a lower cost in a different country. Regardless of how offshoring is accomplished, there are many challenges that are inherent in the process. These challenges include the following: differences in language and culture, lack of domain knowledge about US companies for foreign workers, and lack of experience in systems development for foreign workers. This paper examines these challenges and proposed solutions to them from current research. By analyzing the proposed solutions from the perspective of a SME with a captive development office in India, it is found that too many suggested solutions are either naïve or incomplete and do not fully address all the challenges in an informed way. Further studies are suggested to find better solutions to the challenges involved in offshore software development.

INTRODUCTION:

"Offshoring can be defined as the relocation of business processes (including production, distribution, and business services, as well as core activities like research and development) to lower-cost locations outside national borders" (Erber and Sayed-Ahmed, 2005). Offshoring business processes occurs frequently within the context of outsourcing; however, a process need not be outsourced to be offshored. Many US firms create subsidiaries in foreign countries to perform specific business processes. This practice is called going captive (Kobayashi-Hillary, 2005).

Although the offshoring of information technology (IT) is a relatively new practice, outsourcing and offshoring are not new. Outsourcing in the US is as old as the industrial revolution (Kakumanu and Portanova, 2006). Companies do not make their own paper; they do not produce their own electricity; they depend on other companies that can provide these products and services at a lower cost and a higher level of quality. Outsourcing became especially popular in the US during the 1990s when businesses began to focus more on their core

competencies (Erber and Sayed-Ahmed, 2005). The prevalent idea for US businesses right now is that outsourcing is strategic for functions that are not core competencies (Kakumanu and Portanova, 2006). Offshore outsourcing has been a common practice for many US firms. Over the last 30 years, textile, automobile, and steel manufacturers have outsourced to foreign countries (Erber and Sayed-Ahmed, 2005).

The offshoring of IT has only become popular in the last 10-15 years. The combination of the dot com boom coupled with the Y2K problem created demand for IT workers that exceeded the US workforce. Therefore, US firms began to look to foreign countries to find skilled workers to fill the need (Kakumanu and Portanova, 2006). With the infrastructure provided by telecom and the ease of access to the internet, it became relatively easy to transfer work performed in a foreign country to the US. Then the dot com bust and the resulting economic slowdown in the US caused businesses to reduce their IT budgets. Companies that wanted to continue developing and supporting their computer systems began to send more work overseas, taking advantage of talented IT workers at a much lower cost (Kakumanu and Portanova, 2006).

Many large US firms are continuing to increase their levels of offshoring (Weier, 2007). However, not all IT companies are engaged in the process (CIO Update Staff, 2008). And many companies that were offshoring software development have ceased those operations.

CHALLENGES:

So why are more US companies not taking advantage of the less expensive IT workforce in foreign countries? One reason is the fact that there are many challenges inherent in the process of offshoring IT work. For the sake of organization, this paper groups these challenges into the following categories.

LANGUAGE AND CULTURE

Language and cultural differences contribute to many problems in IT offshoring (Berenbach, 2006; Bertch, 2003; Bhat, Gupta, and Murthy, 2006; Borchers, 2003; Cady, 2005; Damian and Zowghi, 2002; Daruvala, 2003; Erber and Sayed-Ahmed, 2005; Ford, Connelly, and Meister, 2003; Gurung and Prater, 2006; Kakumanu and Portanova, 2006; King, 1998; Kobayashi-Hillary, 2005; Krishna, Sahay, and Walsham, 2004; MacGregor, Hsieh, and Kruchten, 2005; Matloff, 2005; McLaughlin, 2003; Meall, 2002; Mikulovic and Heiss, 2006; Murray and Crandall, 2006; Narayanaswamy and Henry, 2005; Overby, 2004; Overby, 2006; Overby, 2007; Thurm, 2004). Although most IT workers in foreign countries speak English, there are still problems with communication (Cady, 2005; Gurung and Prater, 2006; Kakumanu and Portanova, 2006; Kobayashi-Hillary, 2005). Accents and style can affect basic understanding. Workers in other countries often use words and phrases incorrectly (Gurung and Prater, 2006). In some cases, they blend their own language with English. "You may be baffled as *dacoits* rob *wallahs* of *lakhs* and *crores* in this strange pidgin 'Hinglish.' Engineers tell their perplexed manager they are working late because of the software 'upgradation' or system 'updation'" (Kobayashi-Hillary, 2005, p. 58-59). The misunderstandings caused by the differences in language can cause animosity between American and foreign workers (Gurung and Prater, 2006).

In addition to language, there are many cultural differences that affect the success of IT offshoring (Cady, 2005; Gurung and Prater, 2006; Kakumanu and Portanova, 2006; King, 1998; Kobayashi-Hillary, 2005; Matloff, 2005; Murray and Crandall, 2006; Overby, 2004; Overby, 2007). Many foreign cultures, particularly those of India and China, value well defined hierarchical relationships (Borchers, 2003; Ford et al., 2003; Kobayashi-Hillary, 2005; MacGregor et al., 2005; Matloff, 2005; Narayanaswamy and Henry, 2005). Therefore, communication is expected to flow down certain channels, and it can be confusing to foreign workers when Americans ignore this social structure. This hierarchical structure also can cause foreign workers to withhold relevant information. Julia King (1998) relates an example of this problem:

In one case, user testing failed even after an elaborate system check had been conducted by the offshore IT staff. When a Sprint IT director turned to the vendor and said, "I thought you told me you tested it," the vendor responded that he had. But he added that the IT director had never asked him whether the test had been successful. (p. 2)

The educational systems in many of these countries have different values as well. While American educational systems value creativity and initiative, Indian and Chinese educational systems reward following strict directions without questioning the authority of educators. The result of this educational culture is workers who "can't think independently or creatively, and who can't solve practical problems" (Matloff, 2005). Wesley Bertch (2003) points out that there can be an overemphasis on process:

I never thought I would say that an offshore vendor is too process-dependent. I had always listed this vendor's quality and process focus as a strength – and it can be. But process by itself can't assure project success, and documentation can't substitute for domain expertise.

Like a contract manufacturing plant, the offshore model is designed to funnel any and all projects through a labyrinth of processes and internal controls so that novice employees who don't know anything about a customer's business can achieve acceptable results.

The problem is that you can't factory-produce this kind of software. Developing software is more like team surgery, where competency, experience, group chemistry and knowledge of the patient go a lot further than a set of processes for how the surgery should be performed. (p. 70)

Communication styles and attitudes toward authority can also differ between cultures (Cady, 2005). In many cultures, workers are accustomed to getting detailed directions for projects. However, most American software development companies give only general directions for what is to be developed, and programmers have the freedom to achieve the solution in the manner that seems best to them. Therefore, many foreign workers are seen as too dependent on the people giving direction (Kobayashi-Hillary, 2005). Stephanie Overby (2007) interviewed two experts in IT outsourcing, Mary C. Lacity and Joseph W. Rottman. In this interview, Rottman explained this problem well:

A good example is that you'd hear a US manager complain, "You really have to give the offshore development teams very specific instructions. They really don't show much initiative in going out and investigating a new solution to a problem." And when you talk to someone on the supplier side, they would say, "The US managers don't give us all the information that we need. They don't direct us. We're kind of left on our own." (p. 4)

BUSINESS DOMAIN KNOWLEDGE

Just because a software developer in a foreign country knows how to program, that does not mean the developer will be able to solve a problem for a US business by developing software. The reason for this is that developing software systems to support businesses requires two types of knowledge – programming knowledge, and business domain knowledge. Programming knowledge can be learned through books, but it is refined through practice. In the same way, a basic level of business domain knowledge can be learned by reading documentation, but a deeper understanding can only be achieved through experience. Since most foreign IT workers do not have experience working in US businesses, they lack the business domain knowledge that is necessary to develop software systems (Berenbach, 2006; Bertch, 2003; Cady, 2005; Kakumanu and Portanova, 2006; Kobayashi-Hillary, 2005; Mikulovic and Heiss, 2006; Murray, 2006; O'Sullivan, 2008; Overby, 2004; Overby, 2007; Thurm, 2004). In the experience of Robert Cady (2005), "Lack of domain expertise and deep understanding of the client's pains and corporate culture limited the ability of an offshore vendor from delivering an effective solution" (p. 53). This problem is not as well documented as others, but it might be the most significant problem relating to IT offshoring. If offshore workers do not understand US business problems, it will be extremely difficult to develop software systems to solve these problems.

EXPERIENCE

IT workers in most foreign countries lack experience in systems development (Bertch, 2003; Cady, 2005; Gopalakrishnan, Kochikar, and Yegneshwar, 1996; Kakumanu and Portanova, 2006; Matloff, 2005; Overby, 2004; Overby, 2006; Overby, 2007; Thurm, 2004). In India, programmers average about two year of experience (Bertch, 2003; Matloff, 2005). Scott Thurm (2004) described the process an American company went through to outsource a particular software development project: "US executives wanted programmers with eight to ten years of experience, typical of ValiCert's US employees. But such 'career programmers' are rare in India, where the average age of engineers is 26. Most seek management jobs after four or five years" (p. A.1). This finding is echoed in an account of a different American company recounted by Wesley Bertch (2003):

Indian software labor is highly educated and dedicated, to be sure, but we found that workers lack the technical and people skills that come only with experience.

Our vendor's employees averaged only two years' experience. Because so much was riding on this trial project, the vendor assigned us a "senior" team: The Java and JSP developers each had four years of experience, and the tester had two years of experience. By comparison, any one of our internal Life Time software developers has more experience than the entire offshore team combined. (p. 67)

Lack of experience on systems development projects can have devastating effects. As with most skills, systems analysis, systems design, and programming are developed by practice. Therefore, there is no substitute for experience. Inexperienced programmers can take much more time to develop software, with a much higher error rate in the code, if they can develop the software at all. This can result in suboptimal programs and failed projects.

In addition to lack of experience, many IT businesses in foreign countries experience high turnover (Cady, 2005; Kakumanu and Portanova, 2006; Overby, 2006). This compounds the problems caused by lack of experience because new employees must be continually retrained. It makes long projects especially difficult to complete because there is a great deal of time involved in understanding the project itself.

POSSIBLE SOLUTIONS:

The challenges that have been documented concerning IT offshoring are indeed significant. In response to these challenges, a few researchers have discussed possible solutions (Bertch, 2003; Cady, 2005; Daruvala, 2003; Erber and Sayed-Ahmed, 2005; Gopal, Mukhopadhyay, and Krishnan, 2002; Gurung and Prater, 2006; Kakumanu and Portanova, 2006; Kobayashi-Hillary, 2005; Krishna et al., 2004; Matloff, 2005; Murray and Crandall, 2006; Overby, 2007; Shao and David, 2007; Thurm, 2004).

Wesley Bertch (2003), director of software systems at Life Time Fitness, looks back on a failed offshore project:

What would we do differently? Instead of relying on the vendor to institute the offshore processes and team, we would set that up ourselves. Ideally, we would have a developer (probably an Indian) from our internal team relocate to India to build and manage a competent offshore team, perhaps within leased space at an existing development facility. (Bertch, 2003, p. 67)

Although not many researchers suggest that companies should go captive, many advise taking more control of projects from the US location (Bertch, 2003; Kobayashi-Hillary, 2005; Matloff, 2005). This can take many forms. Members of the offshore team can visit the US location. Members of the in-house project team can visit the offshore location. Krishna, Sahay, and Walsham (2004) advise using people who bridge cultures on offshored projects. The important part of this is that there is coordination and communication between both teams (Kakumanu and Portanova, 2006; Matloff, 2005).

A common piece of advice given by a few researchers is to use a consultant or go-between to help with the whole offshoring process, from selecting a vendor to overseeing the contract (Cady, 2005; Matloff, 2005). This can aid companies that do not have the experience or knowledge necessary to begin using offshore development teams. Whether an outside consultant is used or not, though, Cady (2005) asserts that it is wise for a company to take steps to understand the offshoring process. He quotes an offshoring consultant, Narsi Narasimhan:

Offshore today is a matter of education for US corporations. There is a culture gap and simply jumping offshore is risky without understanding that market and without organizing to manage far off contractors. Someday, offshore will be business as usual, but now it's relatively new, especially for industries that are known to be conservative by nature. Many companies will need help understanding how to work with this new mode of operation to get the best results. (Cady, 2005, p. 55)

To mitigate risks involved with specific offshore vendors, some researchers suggest vendor diversification (Erber and Sayed-Ahmed, 2005; Overby, 2007). They say that a project can be divided into subprojects. Then each subproject can be given to a different offshore vendor for development. The benefits of such an approach are obvious. If one of the subprojects fails, it can be reassigned to a different vendor, saving a huge amount of time and resources when compared with the failing of the entire project. Another benefit of this approach is increased security. Third party vendors seeking to sell a company's proprietary software can do little with only portions of the proprietary software system.

Many of these responses require very close control over the development process. Therefore, companies involved in IT offshoring need to have good process discipline (Kakumanu and Portanova, 2006; Murray and Crandall, 2006).

To outsource effectively, a company needs good process discipline. Most companies enter into outsourcing agreements without that. This can lead to escalating costs, poor results, and difficulties managing the relationship. The elimination of easy, clear communications channels can further complicate the development process. Process discipline can be lax within companies because employees have the knowledge and communications channels ("over the cubicle") to overcome these lapses. Once a process is outsourced, this knowledge and communication can be lost, causing poor results. (Kakumanu and Portanova, 2006, p. 4)

Scott Thurm (2004) investigated one company's experiences with using offshore workers to help with the development of IT projects. In the beginning, these workers were used in the same way that equivalent employees were used in the US offices. However, through the company's experience, they learned that it was better to assign the offshore team bigger projects, "rather than tasks requiring continual interaction with US counterparts" (p. A. 1). Eventually, this company reorganized many positions within the US offices to better use the offshore labor. An example of this is a software architect who coordinates the work of the US and offshore teams. "That often means exchanging e-mail from home with engineers in India between 11 p.m. and 3 a.m. California time, as Mr. Haines reviews programming code and suggests changes" (p. A. 1). That reorganization of teams and positions allows the company to realize the full potential of employees in both locations.

Managing cultural difference between development teams is important to the success of an IT offshoring project (Gurung and Prater, 2006; Kobayashi-Hillary, 2005). "A collaborative culture that includes trust and openness between outsourcing partners is important for outsourcing success. To bridge cultural gaps, companies may offer cultural training, cultural awareness

initiatives and team-building workshops" (Gurung and Prater, 2006, p. 30). These steps can help prepare employees to communicate effectively with members of another culture.

A related idea was expressed by Mary Lacity, in an interview conducted by Stephanie Overby (2007):

Work gets done by people. It doesn't get done by processes. It doesn't get done by documentation. If you want to put an umbrella over all the things that we've talked about – an onsite engagement manager, job shadowing, bringing over a CMM expert from overseas – it's really about building the social capital between the customer and the supplier. (p. 6)

Several researchers used insights gained from studies of cultural differences reported by Geert Hofstede to respond to cultural problems in global software development (Borchers, 2003; Narayanaswamy and Henry, 2005; MacGregor et al., 2005; Ford et al., 2003). Greg Borchers (2003) was on a team of US software developers collaborating on projects with teams in Japan and India. After reading a book by Hofstede during a plane flight from India back to the US, Borchers (2003) recognized how the cultural indexes outlined by Hofstede explained many of the problems that he had experienced during the software development projects. By understanding the cultural index scores of different cultures, Borchers (2003) was better able to collaborate with the teams in Japan and India.

DISCUSSION:

This researcher has worked as the Director of Operations for a SME with a captive development office in India since August 2006. The company, ABC Tech, develops and supports custom software systems for local businesses. ABC Tech has been developing software since 1998, but the company had a difficult time growing due the difficulty in finding competent developers for the amount that was available for hourly pay. To develop software systems for local businesses, ABC Tech needs to have extremely low costs, because local businesses cannot afford to pay high hourly rates for development. Therefore, the president of ABC Tech moved to India in August 2006 to establish a subsidiary of the company to perform the development operations.

My role in the company has been to manage all phases of projects from the US side. This includes meeting with clients, gathering requirements, communicating the requirements to the offshore team, testing, implementing solutions, and supporting the completed systems. I have experienced first-hand many of the challenges described by researchers. I still have difficulty understanding our team leader when we talk every Monday over Skype. I was initially frustrated by our programmers' need for detailed instructions, but I have learned to explain things better. The time zone difference makes scheduling meetings difficult, and our budget does not allow for many visits. Our programmers have a very hard time understanding the purpose of the systems that we develop, not because they do not understand the language used to describe the processes, but because they do not understand the business domain. The programmers in our office all have less than two years of experience. We tried to find workers with more experience, but so far we have been unsuccessful. One of the clients of ABC Tech is a home health care provider. Because the system we developed for this client stores health care information, we had to go

through extra steps in development to protect the sensitive information. We also experienced many of the hidden costs in the development process. For example, because of the state of the infrastructure in the specific city in India where we operate, we have two different internet service providers and a gas-powered electricity generator.

In addition to the challenges described by researchers, we experienced a few that have not been discussed. For example, when it came time to review the performance of one of our lead programmers, we encountered some unrealistic expectations. This programmer performed well on all of our metrics, and we planned to give him a 20% raise in salary. From our perspective this was a generous amount; we thought he was well paid to begin with, and at the time of this review he had barely two years of experience. His productivity was good relative to other programmers in the office, but he was still only producing about a quarter of what our best US programmer could produce. When he was told that he would receive a 20% raise, he was devastated. Apparently, he expected to receive about twice as much as his current salary when he had two years of experience, as if two years was a magic number. We would be happy to double his salary if he could double his productivity, but that was simply not the case. Stories of what other programmers claim to make are exaggerated, and the result is unrealistic expectations for other programmers.

The possible solutions described by researchers are all important, and it would be good for companies that are exploring the option of offshoring to implement many of the strategies previously documented. However, from the perspective of someone who is performing offshore software development, none of these responses adequately solves some of the major issues with offshoring. For example, how does a company solve the problem of inexperienced programmers? Does that company invest time and money in training programs to get foreign programmers to the same level as US programmers? Would the company also need to provide training in basic understanding of US businesses? Assuming that is even possible (which is not likely), would that company realize any savings at all, especially considering the problems with employee turnover? Does it really make sense to invest resources in social capital, when the entire offshore development team can change within the time frame of one project?

Too many suggested solutions are either naïve or incomplete and do not address fully all the challenges in an informed way. Therefore, better solutions are needed. Following is a suggested process.

First, better research is needed to describe fully how businesses perform offshore software development. This would illuminate all of the challenges involved in the process, and it would reveal how businesses currently manage these challenges. This researcher is planning to perform a grounded theory study of how workers manage the process of offshore software development. Interviews will be conducted with workers in India as well as the US workers who interact with them. The interviews will be analyzed to build grounded theory.

Next, researchers should investigate each of the methods that businesses use to manage the challenges to determine which of them is effective. For example, some businesses might use an expert developer from the US to move to India to be a resource for the Indian development team.

Quantitative analysis could be performed to determine if this practice results in a more efficient and effective process.

Finally, researchers could propose new solutions grounded in an accurate understanding of the overall process. For example, since many businesses experience problems because of the challenge of a lack of domain knowledge for offshore software developers, a researcher might propose a strategy of performing all analysis functions in the US. The effectiveness of this strategy could be tested by quantitative analysis.

In conclusion, there is great potential for cost savings by using offshore IT workers to perform software development. However, because of the significant challenges involved in the process, solutions are needed to make the process successful. A few methods have been proposed by researchers for addressing the challenges, but more research is needed to propose solutions that fully address the challenges in an informed way.

REFERENCES

- Berenbach, B. (2006). Impact of organizational structure on distributed requirements engineering processes: lessons learned. In *Proceedings of the 2006 international Workshop on Global Software Development For the Practitioner* (Shanghai, China, May 23 - 23, 2006). GSD '06. ACM, New York, NY, 15-19.
- Bertch, W. (2003). Why offshore outsourcing failed us. *Network Computing*, 14(21), 65-70.
- Bhat, J. M., Gupta, M., Murthy, S. N. (2006). Overcoming Requirements Engineering Challenges: Lessons from Offshore Outsourcing. *IEEE Software*, 23(5), 38-44.
- Borchers, G. (2003). The software engineering impacts of cultural factors on multi-cultural software development teams. In *Proceedings of the 25th international Conference on Software Engineering* (Portland, Oregon, May 03 - 10, 2003). International Conference on Software Engineering. IEEE Computer Society, Washington, DC, 540-545.
- Cady, R. P. (2005, September). Expertise Required to Reap Rewards From Overseas Outsourcing of IT. *Pulp & Paper*, 79(9), 52-55.
- CIO Update Staff (2008). Overseas IT Outsourcing Rates Remain Low. *CIO Update*, January 25, 2008. <http://www.cioupdate.com/research/article.php/3723981>.
- Damian, D. E. and Zowghi, D. (2003). An Insight into the Interplay between Culture, Conflict and Distance in Globally Distributed Requirements Negotiations. In *Proceedings of the 36th Annual Hawaii international Conference on System Sciences (Hicss'03) - Track1 - Volume 1* (January 06 - 09, 2003). HICSS. IEEE Computer Society, Washington, DC, 19.3.
- Daruvala, T. (2003). When, Where, How, and Other Questions on Going Offshore. *American Banker*. 7/3/2003, Vol. 168 Issue 127.

- Erber, G., Sayed-Ahmed, A. (2005). Offshore Outsourcing: A Global Shift in the Present IT Industry. *Intereconomics*, 40(2), 100-112.
- Ford, D. P., Connelly, C. E. and Meister, D. B. (2003). Information Systems Research and Hofstede's Culture's Consequences: An Uneasy and Incomplete Partnership. *IEEE Transactions on Engineering Management*. Volume 50, Issue 1, p. 8-25.
- Friedman, T. (2005). *The World Is Flat: A Brief History of the Twenty-first Century*. New York: Farrar, Straus and Giroux.
- Gopal, A., Mukhopadhyay, T., and Krishnan, M. S. (2002). The role of software processes and communication in offshore software development. *Communications of the ACM*. 45, 4 (Apr. 2002), 193-200.
- Gopalakrishnan, S., Kochikar, V. P., and Yegneshwar, S. (1996). The offshore model for software development: the Infosys experience. In *Proceedings of the 1996 ACM SIGCPR/SIGMIS Conference on Computer Personnel Research* (Denver, Colorado, United States, April 11 - 13, 1996). SIGCPR '96. ACM, New York, NY, 392-393.
- Gurung, A., Prater, E. (2006). A Research Framework for the Impact of Cultural Differences on IT Outsourcing. *Journal of Global Information Technology Management*, 9(1), 24-43.
- Hofstede, G. (1980). *Culture's Consequences: International Differences in Work-Related Values*. Beverly Hills, CA: Sage Publications.
- Huckvale, T. and Ould, M. (1995). Process modeling: who, what and how, in Grover, V. and Kettenger, W.J. (Eds), *Business Process Change, Reengineering Concepts, Methods, and Technologies*. Hershey, PA: Idea Group Publishing.
- Janamanchi, B., Burns, J.R. (2007). Offshoring Knowledge-worker Jobs: boom or Burst for the US Economy? *The Icfai Journal of Knowledge Management*, Vol. V, No. 2, p. 20-39, Mar.
- Johnson, C. (2007). Outsourcing Your Way to Innovation: IT Leaders Question the Value. *CIO.com*, September 26, 2007, http://www.cio.com/article/141350/Outsourcing_Your_Way_to_Innovation_IT_Leaders_Questi_on_the_Value.
- Kakumanu, P., Portanova, A. (2006). Outsourcing: Its Benefits, Drawbacks and Other Related Issues. *Journal of American Academy of Business*, Cambridge, 9(2), 1-7.
- Kettinger, W.J., Teng, J.T.C., Guha, S. (1997). Business Process Change: A Study of Methodologies, Techniques, and Tools. *MIS Quarterly*, Vol. 21, No. 1, pp. 55-80.
- King, J. (1998, May). Benefits can be elusive; costs are deceptive. *Computerworld*, 32(20), 1, 16.
- Kobayashi-Hillary, M. (2005). A passage to India. *ACM Queue*, 3(1), 54-60.

Krishna, S., Sahay, S., and Walsham, G. (2004). Managing cross-cultural issues in global software outsourcing. *Communications of the ACM*. 47, 4 (Apr. 2004), 62-66.

Kruchten, P., Hsieh, Y., MacGregor, E., Moitra, D., Strigel, W., and Ebert, C. (2006). Introduction to the 1ST international workshop on global software development for the practitioner. In *Proceedings of the 2006 Int'l Workshop on Global Software Development For the Practitioner* (Shanghai, China, May 23 - 23, 2006). GSD '06. ACM, New York, NY, 1-2.

Lacity, M. (2002). Lessons in Global Information Technology Sourcing. *Computer* 35, 8 (Aug. 2002), 26-33.

FOR REMAINING REFERENCES, CONTACT THE AUTHORS