

Offshoring: What Can Go Wrong?

Norman Matloff

Comedian Woody Allen said, “Eighty percent of success is showing up.” On the topic of offshoring, that says it all. Humans work best in physical proximity to each other. Yet today, global differentials in pay rates make it tempting for companies to employ people working

to offshore its IT work—specifically software development and maintenance? I’ll discuss several reasons for firms to be wary of this practice. For some managers, these considerations might tip the balance against offshoring. For those who wish to offshore anyway, this article can serve as a map of the pitfalls to try to avoid.

THE VALUE OF FACE TIME

Proponents of offshoring blithely say that companies can overcome any geographical obstacles by using the proper tools. But no matter how advanced your telecommunications tools are—videoconferencing, software for managing collaborative development, and so on—it’s still not the same as being there.

Indeed, the work models of vendors from India illustrate this point. Although the bulk of

a project’s team might be in India, the offshore firm typically stations some team members onshore—that is, at the client’s site. A University of Rochester study found that the typical ratio is around one onsite worker to every two offshore workers (Ron Hira, “U.S. Immigration Regulations and India’s Information Technology Industry,” *Technological Forecasting and Social Change*, vol. 71, no. 8, Mar. 2005). Some onsite workers remain at the client site for the duration of the project; others make temporary visits for training or consultation.

Craig Hergenroether, CIO of Barry-Wehmiller Companies, is both a consumer and purveyor of offshore services; his US firm has a subsidiary in India. His answer to my query about the importance of physical proximity illustrates the point in more detail:

Distance, cultural differences, inexperienced programmers, and other obstacles might make you wish you’d kept that IT project at home.

tens of thousands of miles from the home base—a client in the US offshoring IT work to a vendor in India or China, for example. Can it be done well?

Offshoring has recently sustained much criticism, mainly over the issue of whether shipping IT work abroad is good for the US as a whole. I have written in that context as well, but my focus here is at the company level. Does it benefit a US firm

Inside

Of the all the issues in IT today, outsourcing is indisputably the most controversial. Here, the author offers his views on why companies should carefully consider the choices they make in offshoring. A professor of computer science, Norman Matloff has written widely on work visa issues and testified before the US Congress about their effects on the IT workforce.

As always, *IT Pro* welcomes and encourages opposing viewpoints; contact us at itpro@computer.org.

—Frank E. Ferrante, Editor in Chief

... any significant project related to development must be accompanied by onsite interaction. This is true whether it is outsourced in Chicago or Chennai. Phone conversations are fine for straightforward conversions or any other pure labor endeavor. When it comes to development work, you must understand the standards and local nuances of the client. ... This interaction can take place over several weeks and then move back offshore. However, we have found that the communications will deteriorate over time if there is not periodic face-to-face interaction.

Clearly, then, even firms that provide offshore services believe that physi-

cal presence onsite does matter.

Ed Frauenheim, a journalist who covers the tech industry, once spent a day shadowing an engineer at a Silicon Valley company. He was surprised to see how much work was done serendipitously, in the office hallways. This should come as no surprise to any IT manager. Good software development requires constant informal interaction among developers and managers, the ability to go down the hall to talk face-to-face on the spur of the moment.

Because of the time differences between the US and the chief offshoring service countries, typically the best interaction offshoring can offer is a discussion every 24 hours via

videoconferencing; this isn't enough. If a question arises suddenly, you wait a whole day for the next meeting. And because most people are much more comfortable meeting in person, they're more likely to ask all their questions and elaborate on all the issues when they're sitting on opposite sides of a table, instead of opposite sides of the globe.

THE QUALITY GAP

A key but not widely known point about offshoring is that its typical business model minimizes costs by staffing projects with young, inexperienced programmers. The median age of programmers in India is 25.6 (En Interactive Technologies, <http://www.eninteractive.com/Why+Choose+India>). Youth is the salient characteristic of the onsite workers as well. At giant Tata Consultancy Services, headquartered in Mumbai, India, 50 percent of the H-1B visa programmers (the workers who travel to client sites) are under age 25, and 88 percent are under 30 (TCS recruiting Web page, "Why Join TCS America," 2004).

Arvind Thakur, president of the offshoring firm NIIT, has explained that the philosophy underlying this model is to have higher-level, more experienced personnel divide a programming project into small pieces that less-experienced programmers can easily handle. The simplicity of the individual tasks is supposed to neutralize the programmers' inexperience. In my observation, however, projects rarely attain the ideal of the easy pieces, leaving a programmer's relative lack of experience to adversely affect quality.

As evidence that they can solve such problems, many offshoring firms in India would point to their adherence to the highest standards of the Capability Maturity Model (CMM, <http://www.sei.cmu.edu/cmm>). But CMM wasn't designed as a substitute for experience. Let's take a closer look.

CMM is a checkpointing system

What about Linux?

You might wonder whether the Linux operating system provides evidence that offshoring can pay off. I had often wondered about this point myself, so I put the question to Linus Torvalds, founder of the Linux project. Torvalds replied that the two models of software development aren't comparable:



I don't think the Linux model works for offshoring in the commercial sense, or really ends up even being very relevant. The problem ends up being communication and the mental model pretty inherent in offshoring.

My belief is that when you say "offshoring," you very much mean "control the project on one shore; work on the other." That is, the implication of the offshore work being "subservient" is very much there in the notion of offshoring.

In contrast, the Linux model (and open-source in general) is that there's no one-sided control, and that when work gets done overseas, it gets done because it makes sense to them, not to "us." There's no control of one end over the other—both shores do what they want to do. The fact that makes it all work out is that, in the end, everybody tends to have somewhat overlapping goals.

Of course, the goals don't always overlap. Although all Linux distributions use the same kernel, at least two major user interfaces have evolved—KDE and Gnome. So again, this differs greatly from the offshoring context. Another major difference is that Linux, as a noncommercial product, is developed pretty much without time pressure. The next version is ready when it's ready. In most commercial settings, this leisurely approach would be out of the question.

that's supposed to put management of a software project on a rigorous basis. A company can seek CMM certification at quality levels up to level 5—and now even beyond, with certifications such as the new Team Software Processes assessment. The Indian offshoring industry has made CMM its marketing cornerstone. Currently, more than half of the firms worldwide that are certified at CMM level 5 are in India.

However, most of the commotion about CMM is misleading. The system does impose a degree of discipline on the software development process. But what companies overlook in all the certification enthusiasm is that CMM is simply a project management tool—not a measure of software quality. Even Jay Douglass, an executive at the Software Engineering Institute (SEI) at Carnegie Mellon University, home of the CMM, has pointed out, “You can be a level-5 organization that produces software that might be garbage.”

In fact, although SEI (hardly an unbiased source) claims that level-5 firms have code error rates sevenfold lower than level-1 firms, a study of 89 level-5 companies by Reasoning, a software quality consultancy, found that, on average, the code produced by level-5 firms had higher error rates than those of non-CMM-rated firms (Christopher Koch, “Bursting the CMM Hype,” *CIO*, 1 Mar. 2004, <http://www.cio.com/archive/030104/cmm.html>).

A quality assurance specialist for a major US firm bluntly told me her negative experience with trusting CMM: “We used [a major CMM level-5 firm] on several projects. Not one line of their code made it into either our test suites or finished products.” The fact is that many experts consider CMM, at least at the higher levels, to be bureaucratic overkill; software developers end up spending far more time on documentation than is beneficial.

According to an article in *BusinessWeek*, programmers at Indian

offshoring companies tend not to be as productive as their US counterparts at the client company (David Gumpert, “An Unseen Peril of Outsourcing,” 3 Mar. 2004). The article quotes one veteran offshoring client as saying, “I felt we needed one-and-a-half times the Indian programmers to do the same amount of work as US programmers.”

Another quality issue is that the client seldom vets the selection of the offshore programmers.

Another offshoring client quoted in the article put the ratio at about five or six to one (Indian to US programmers). As an Indian-American developer explained to me, it's part of the culture: “In India, there is overmaning wherever you go.”

The fact that the offshore programmers tend to be rather recent graduates has another implication: New graduates of non-US universities might be “greener” than those educated in the US. I asked an undergraduate exchange student of mine from Hong Kong University—one of the most selective universities in Asia—to describe the difference between her courses there and those she took this year at the University of California, Davis. She noted that her classes in the US involved much more discussion, with students being asked to evaluate alternative approaches to solving a problem. “Here [in the US], class time is much more interactive. At HKU, we just sit there.”

Though new US graduates might not be conversant in a wide variety of software tools, many are arguably more ready than their offshore peers to engage in the constant decision making necessary in the software development process. As one ethnic-Indian programmer told me, “The

American employees are more assertive, offering solutions, more proactive.”

Project manager Wesley Bertch, who wrote about his offshoring experiences for *Network Computing* (see the “Further Reading” sidebar), has also described the consequences of using inexperienced programmers:

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 ... developers has more experience than the entire offshore team combined. ... This development inexperience led to a series of rookie blunders.

These eventual blunders occurred even though the offshore vendor in that case was from the so-called tier 1 group, composed of the top firms in India.

Further Reading

- ▶ Wesley Bertch, “How Offshore Outsourcing Failed Us,” *Network Computing*, 16 Oct. 2003, <http://www.nwc.com/showArticle.jhtml?articleID=15201900>.
- ▶ Deloitte Consulting, LLP, “Calling a Change in the Outsourcing Market,” 19 Apr. 2005.
- ▶ Christopher Koch, “Bursting the CMM Hype,” *CIO*, 1 Mar. 2004, <http://www.cio.com/archive/030104/cmm.html>.
- ▶ Stephanie Overby, “The Hidden Costs of Offshore Outsourcing,” *CIO*, 1 Sept. 2003, <http://www.cio.com/archive/090103/money.html>.

In a 1999 article in the American Society for Engineering Education's *Prism* magazine, an engineering professor in China, Chen Lixin, warned his nation that the engineers coming out of Chinese universities weren't good enough to make China competitive in the global high-tech market ("Reform: China's New Engineering Obstacle," <http://www.prism-magazine.org/sept99/lastword.htm>). Chen complained that the Chinese educational system produces students who can't think independently or creatively, and who can't solve practical problems. He wrote that the system "results in the phenomenon of high scores and low ability."

Another offshoring quality issue is that the client seldom vets the selection of the offshore programmers. As one US developer put it to me, at his firm, "we would never hire anyone like [those offshore programmers] here, but we didn't get to do any screening. Their experience was in system integration, yet they were assigned to work on our large, complex application development project."

LANGUAGE AND CULTURE

An old story in China tells of a Mandarin-speaking official in Beijing who ordered a Cantonese-speaking worker to inspect some telegraph

wires. The latter thought he heard *dismantle* rather than *inspect*, and alas, complied. Thus the problem of linguistic mismatch in technology projects is not a new phenomenon, though it tends to take more subtle forms today. In India, today's dominant offshoring site, many people have been speaking English since age five, if not earlier. In fact, many Indian engineers put their US counterparts to shame in terms of working vocabulary, richness of expression, and so on. But Indians speak their own dialect of English, with pronunciations and idioms that can sometimes baffle their coworkers in the US; imperfect audio clarity on the videoconferencing link can worsen the situation. Spending some extra effort to communicate might not seem overly burdensome, but some reports have described language problems working their way into the product or into the code itself. In source code comments, the phrasing an offshore team uses can seem odd, even downright cryptic. This can compromise code maintainability: a hidden cost of offshoring that surfaces later. One frustrated developer described offshoring a GUI project: the Chinese team that worked on it used completely unacceptable phrasing, which then had to

be redone.

GUI work can encounter cultural obstacles as well. Say, for instance, a project involves a GUI that lets a company's employees check their 401(k) accounts; this is literally a foreign concept to some non-US programmers. The result could be a GUI that consumers don't find comfortable, or even usable. The need to correct problems like these is another typical hidden cost of offshoring. A client firm might avoid such problems by providing extremely detailed specifications, but that too would mean extra time, thus extra cost in some form.

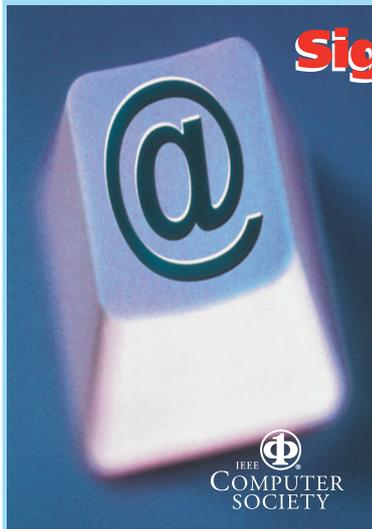
Cultural differences can also affect the work process. For example, the typical US workplace informality can clash with the hierarchical structures common in India and China. A US programmer communicating directly with her peer in India, bypassing upper management layers, might inadvertently cause resentment.

Indian workers' stricter observance of workplace hierarchies can also make them overly reluctant to express their opinions; some companies have reported offshore teams following a client's program specifications even though the team saw that the specs wouldn't work. Similar problems might arise in China, with its Confucian traditions of respect for authority.

WORKING ROUND THE CLOCK

Another claimed virtue of offshoring is that companies can exploit time zone differences to shorten product development times. With teams in the US and India, one group programs while the other sleeps—such a deal!

When this works—and in some cases it does—it does indeed speed up the development process, a major consideration for some customers. There is, however, the potential for disaster, or at least inefficiency. The situation recalls a scene from the movie comedy *The Odd Couple*, in which the two incompatible roommates alternately move a chair from one room to another and back again, each unaware



**Sign Up Today
for the IEEE
Computer
Society's
e-News**

Be alerted to

- articles and special issues
- conference news
- registration deadlines

**Available for FREE
to members.**

computer.org/e-News

that he is repeatedly undoing the work of the other. This movie should be required viewing for any IT manager tempted by an offshoring firm's claims of a shortened product cycle. One unwilling "Felix Unger" described his offshoring experience to me:

Due to being many time zones away, there is no time to loop back and fix problems or make changes. If the work were onsite, one could just walk over and talk to the lead in five minutes. But with the developers in India, every time you have to talk, it's another day. So, the development cycle is elongated, not compressed. As a result of being able to talk just once a day, the remote developers start working on their own, and start making poor decisions.

Indeed, if the program-around-the-clock model were so good, all US firms would already be using it. Lots of programmers have nighttime personalities; why not assign a few of them to the graveyard shift and have them alternate work with a daytime crew? The fact that this hasn't become a standard model speaks volumes.

One activity that might better fit the around-the-clock model is software testing. It might make sense for testers to do their work in India while US programmers sleep, providing the latter with a new set of bugs to work on when they arrive at work in the morning.

INNOVATION

Offshoring boosters say, "The forté of the US is innovation. It should concentrate on that and offshore the mundane work." Is this a reasonable analysis? If so, what are the implications from a company's perspective?

Technology journalist John Dvorak once pointed out to me that there have been no "killer apps"—dazzlingly creative applications software—from India or China. I can think of one exception (albeit a little-known one), the JiaJia distributed shared-memory library from the

Chinese Academy of Sciences, which has some nice, innovative features. On the whole, though, I agree with the notion that neither country has produced much interesting software.

The governments of several East Asian countries—China (and Hong Kong), Japan, Korea, and Taiwan—believe that their educational systems have traditionally stifled creativity and independent thinking ("Roll Over, Confucius: Reforming Education in China," *The Economist*, 23 Jan. 2003). They have taken steps to reverse the pattern, but this will take decades,

especially because it is as much a cultural issue as an educational one. The emphasis on authority found in Confucianism and other similar traditions means that students find it difficult to question first their teachers, and later established expertise in the working world; this is definitely not conducive to thinking outside the box. On top of that, the overly regimentative CMM system has the effect of stifling innovation.

Accordingly, for projects in which innovation is important, you might avoid sending programming work off-

Consider the Source

A rather alarming aspect of the coverage of offshoring in the popular press is that the reader rarely hears of the vested interests of some of offshoring's proponents.

Consider, for example, the much-touted McKinsey study ("Offshoring: Is It a Win-Win Game?" Aug. 2003, http://www.mckinsey.com/mgi/publications/win_win_game.asp).

A March 2004 article in the *San Francisco Chronicle* noted that "an oft-cited study from consulting firm McKinsey & Co. says that every dollar US companies spend on offshoring will return up to \$1.14 to the domestic economy" ("Tech Association Urges Congress to Delay Protectionist Measures," 24 Mar. 2004). What the article doesn't mention is that this consulting firm sells offshore services.

The Information Technology Association of America (ITAA), a powerful industry lobbying group, commissioned a respected research firm to conduct a study of offshoring, and then widely publicized the study's positive conclusions. Yet the ITAA essentially withheld the study itself from the press and the public, so consumers have no idea what negative conclusions the study's authors might have also reached. Nor did the ITAA disclose how the authors conducted the study, how reasonable their modeling assumptions were, what caveats they made, and so on.

Then there is the Capability Maturity Model itself. Because CMM is a product associated with Carnegie Mellon University (CMU), the studies promoting its usefulness might seem to be free of any vested-interest bias. Sadly, nothing could be further from the truth. Research is a big profit center for enterprising "university.com" institutions. The Software Engineering Institute at CMU reaps \$200 million a year for conducting these studies.

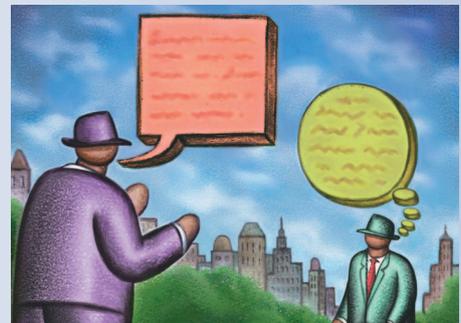


Table 1. Actual reported savings from offshoring.

Savings percentage	Source	Web site
10 to 44	Howard A. Rubin and Patricia Jaramillo	"Outsourcing: An Analysis of the Current State of Offshore Outsourcing in New York City Based Companies," New Jobs for New York, 2004, http://www.newjobsforny.org/OutsourcingReport.php
15 to 40	Sand Hill Group	http://www.sandhill.com
20 to 40	CIO magazine	"The Hidden Costs of Offshore Outsourcing," 1 Sept. 2003; Stephanie Overby, http://www.cio.com/archive/090103/money.html
25 to 50	Gartner Group	Ed Parry, "Gartner: Offshore Outsourcing Horse Has 'Left the Barn,'" 8 Jul. 2003, SearchCIO.com; http://searchcio.techtarget.com/qna/0,289202,sid19_gci913695,00.html .
10 to 20	DiamondCluster International	"India's Salary Growth Threatens Outsourcing," Mike Yamamoto, <i>ZDNetUK</i> , 4 June 2004; http://insight.zdnet.co.uk/specials/outsourcing/0,39026381,39150917,00.htm .

shore. It isn't enough simply to keep the high-level software architectural design onsite, having offshore teams do the implementation. Innovation occurs serendipitously, with programmers coming up with creative ideas while they are working on the actual development. If you don't staff your project with creative programmers, you stand to lose much of your project's potential for innovation.

Of course, the notion of culturally determined creativity deals with population tendencies, not individuals. Not all US programmers are innovative, and many immigrant programmers in the US are creative. You should hire people with a demonstrated track record of creativity—earned, perhaps, by having worked at innovative firms or by doing something out of the ordinary during college.

DEVELOPING AND RETAINING GOOD ONSITE MANAGERS

Clearly, offshoring lowers the morale of your in-house programmers, which makes it difficult for you to hire good ones. But it can hurt you at the management level as well. First, it has been traditional to "grow" IT managers by promoting programmers, so that you have managers who

know your company, its products, its clients, and so on. Obviously, the more programming you move offshore, the drier your internal management pipeline will become. Second, managers of offshore projects face frustrations and disruptions in their family lives that can rapidly lead to burnout. You risk losing your best people.

THE BOTTOM LINE: HOW MUCH DO YOU REALLY SAVE?

The popular press touts the fact that Indian programmers tend to make only about one-tenth of US salaries, with the implication being that offshoring will save you 90 percent of your project expenses. In practice, the savings might be more like 20 to 40 percent. Factors that whittle down the 90 percent include

- infrastructure costs—high-speed network connections, videoconferencing software and hardware, travel between vendor and client sites, and so on;
- higher hourly charges to the client for the vendor's onsite workers;
- cost overruns due to communication problems and to vendor pro-

grammers' lower experience levels; and, of course,

- vendor markup.

The vendors themselves, and other proponents of offshoring, agree that actual savings to client firms are far below the 90 percent reported in the popular press. Table 1 shows some actual figures with their sources.

Two of the sources listed in Table 1 are particularly interesting. First, Howard Rubin's and Patricia Jaramillo's 44 percent savings figure is relative to hiring onsite in New York City, which has an extremely high cost of living. They found that if the company had instead moved its site upstate to Syracuse or Rochester, and employed US labor there, it could attain savings close to those of offshoring.

Second, the DiamondCluster International survey is of special relevance as it is more recent. It warns that wages in India have now risen significantly, reducing the cost savings attainable by offshoring. A 2004 survey conducted by the Gartner Group found that 18 percent of the firms surveyed had attained no savings at all by offshoring, and 9 percent had actually experienced an increase in costs

(“Sending Jobs Overseas Isn’t Always Worth It,” *Los Angeles Times*, 11 Apr. 2004).

HOW TO OFFSHORE IF YOU MUST

If you still wish to move part or all of your software work abroad, here are some points to keep in mind:

- Expect only modest cost savings.
- Decide how important US worker retention is to your business.
- Keep creative work in house—from top to bottom, not just architectural design.
- Write your product specifications in extremely fine detail.
- At the beginning of the project, hire a consultant who can warn you about avoidable language and cultural problems.
- Insist on having detailed résumés

on all overseas workers; if possible, interview each one via videoconference.

- Retain an onsite vendor liaison for the project’s duration. Even with the extra expense, this will make life easier for your managers and avoid costly mistakes and delays.
- Encourage the offshore team to be upfront with you regarding potential problems with specs and design issues.
- Take a hard look at intellectual property issues, especially for your core business products.

The last section’s title alludes to a classic book on probability (*How to Gamble If You Must: Inequalities for Stochastic Processes*, McGraw-Hill, 1965) by Lester Dubins, former professor of statistics at the

University of California, Berkeley. I hope I have raised enough concerns in this article to justify the casino metaphor. There are some winners, but many losers—78 percent of the firms surveyed by DiamondCluster decided to end their outsourcing projects early. However, proper management should increase your odds of winning. ■

Norman Matloff is a professor of computer science and a former software developer in industry. He has also written extensively on IT labor issues. Contact him at matloff@cs.ucdavis.edu.

For further information on this or any other computing topic, visit our Digital Library at <http://www.computer.org/publications/dlib>.

Join the IEEE Computer Society online at

computer.org/join/

Complete the online application and

- Take Web-based training courses in technical areas for free
- Receive substantial discounts for our software development professional certification program
- Get immediate online access to *Computer*
- Subscribe to our new publication, *IEEE Security & Privacy*, or any of our 22 periodicals at discounted rates
- Attend leading conferences at member prices
- Sign up for a free e-mail alias—you@computer.org

THE WORLD'S COMPUTER SOCIETY