salon.com

Tuesday April 04, 2004

When offshoring goes bad

Not all trips to India are blessed by Krishna: A case study of outsourcing gone awry.

The job title on Celeste Smith's business card reads "software project manager," but "surrogate worrier" seems just as apt. After all, the moment her bosses go to sleep is the very moment her nightmares start.

Twelve months ago, Smith, who prefers not to give out her real name lest the blowback put her in the swelling ranks of unemployed U.S. engineers, began work on a high-stakes development project. Her orders were explicit: The company, a major Wall Street bank, needed the system, but with the stock market then in the midst of a three-year slide, it also needed to shave costs. Because her department could employ five skilled Indian programmers for the $1,000 it spent each day on a single U.S. programmer, an Indian subcontractor was quickly hired for the back-end work.

Twelve months and 150,000 frequent-flier miles later, the dark circles under Smith's eyes attest to how well the project is doing. What first looked like a novel shortcut has instead evolved into a death march. Every third week, she and her best Indian coders shuttle back and forth between Bangalore and Manhattan, patching up brittle code and patching over brittle emotions.

Looking back, the number of shattered assumptions reads like a management case study.

"I had to explain to them what batch processing was," says Smith, exasperation showing in her voice. "I had to explain what a job dependency was. Totally basic things you'd expect any 26- or 27-year-old American programmer to know, they didn't know."

Now, with her project nearing completion, Smith faces a final irony: Many of the programmers she just spent the last 12 months bringing up to speed will be gone by the time the first customer support calls start rolling in. Eager to leverage their new experience, they are borrowing a page from their 1990s U.S. peers and shopping résumés all over Bangalore. Smith predicts a turnover rate of 20 percent in the next six months and laughs whenever a vice president, CEO or politician uses "outsourcing" and "cost savings" in the same sentence.

"Sure, we saved money on the labor," she says. "But what about the other costs? What about the cost of rewriting the same piece of code 50 times? What about the cost of delaying other projects, the travel and lodging?"

Anger venting, Smith ends with a flourish: "Where did all our savings go when, at the end of the day, we have a piece-of-shit system that'll just need to be replaced in three years?" Where indeed? Like a lot of other things software-related, overseas outsourcing is an attempt to disguise complexity with simplicity. To some, it's a cost-cutting panacea. To others, it's management's latest attempt to screw the American worker. To those caught in the middle -- the chief technology officers and project managers who have to make it work regardless of their political opinions -- it's simply the latest business trend that, like all trends before it, comes weighted with pluses and minuses.

"Outsourcing can provide a pretty big, immediate cost savings, but it can put a big strain on your internal I.T. [information technology] structure," notes Darrel Raynor, the managing director of Data Analysis, an Austin, Texas, company that specializes in project management. "It takes the implicit costs and makes them explicit. Whereas before you could pull a person off of one effort and put them on a new effort, knowing that person was a sunk cost. With outsourcing, you can't do that so easily, because now you're dealing business to business.

A former senior director of software projects management at Rational, a Lexington, Mass., software development company acquired by IBM in 2002, Raynor witnessed up close the early stages of the current outsourcing wave. Like many other project managers, he sees the process as cyclical. Ten years ago, U.S. companies were in the midst of a similar cost-cutting cycle. Using independent contractors instead of payroll employees, companies were able to show the market fresh profits, igniting a six-year wave of unprecedented growth. That growth brought new projects and the money to manage those projects in-house.

"The pendulum always swings back," Raynor says. "[Outsourcing] is a way for management to lock their business units into the current I.T. infrastructure, but once you ink that deal, it becomes difficult to get anything new done."

The biggest difference between 1994 and 2004 in the I.T. world, of course, is the level of Internet connectivity. Whereas 10 years ago only a few businesses and universities boasted high-speed Internet access, businesses from Moscow to Bangalore now use the Internet for bulk data transfer and video teleconferencing.

Suddenly, given the choice between U.S. and overseas talent, managers are learning to mix and match skills in a bid for lower cost and increased flexibility. They're also learning to accept the added disciplinary demands that come with managing a team 10,000 miles away.

"Ten to 20 percent of your project time should be devoted to communicating what you want," says Eric Laughlin, a self-described "serial entrepreneur" who relies on overseas developers to help turn business concepts into reality. For his latest project, an online aggregator site called CompareWirelessPhones.com, he relied on Altoros, a Tampa, Fla., company with a development team in Russia. After getting burned on two previous projects using Indian subcontractors, Laughlin knew to keep the project scope generic enough to diminish the need for post-development support, and to lessen the risk of undermined intellectual property value.

"I had one company take a huge portion of our code," he says, recalling an Indian contract. "They slapped a new front on it and began selling it as a competitive product. That wasn't fun."

All told, Laughlin says he was pleased with his latest offsourcing effort. "We had to do it fast," he says. "It would have been very expensive to do it in the States."

Raynor, now working with California-based Tiburon Software, reports a similar positive experience with Russian programmers. "They don't seem to mind the repetitive aspects, fixing a bug in six similar versions of software, say," he says. "In almost every American instance I've been involved with over 20 years, if you ask somebody to do something more than three times, you get a lot of pushback."

Such observations lead to a second major difference between now and the last outsourcing wave. In the early 1990s, companies weren't still shaking off the after-effects of a record I.T. spending binge. Burned by technology hype during the dot-com era, some top-level managers have turned to outsourcing as a form of economic and political revenge. Even those that still see local development as competitive also see a potential for the lessons gained through overseas development to change the way development is managed within the United States.

"[During the 1990s] people got very, very lazy with writing specs," says Bob Wourms, the director of outsourcing and project management for PM Solutions, a Pennsylvania training firm. "You try to manage a team in India without detailed specs and you never know what you're going to end up with."

Aside from detailed specs, another way to impose discipline is through standards and standard methodologies. Wourms says he's impressed by the number of Indian firms rallying around the "capability maturity model," or CMM, a management tool developed by Carnegie Mellon's Software Engineering Institute in conjunction with the U.S. Department of Defense. The model rates development processes on a 1 to 5 scale. New projects start out at level 1. Mature projects, where the development processes have been well-codified and are easily teachable to new employees, rank higher.

"They're promoting it as a quality issue," say Wourms. "Every spec is developed in the exact same way. They track errors and turnarounds. Companies that use the same firm over and over again actually aren't in too bad a shape."

The first time out can be painful, however. Smith, whose project didn't call for a specific methodology, holds up her own company as an example of what happens when a U.S. client and an Indian contractor both try to break new ground.

"We've failed at every turn," she says. "We didn't get good specifications, which is essential. We didn't have the best data model. It's a hard thing to outsource and, frankly, it would have even been hard to build in-house."

Jay Douglass, manager of business development for the Software Engineering Institute, likens the recent Indian embrace of industrial software development methodologies to the Japanese embrace of quality-management principles in automaking in the 1960s that Detroit didn't fully incorporate until years later. In both cases, he says, you have maturing industries eager to catch up with the U.S.

"[Development models] have been far more readily adopted in countries like China and India than here," says Douglass.

Whether that leads to a parallel outcome -- Indian developers outperforming their U.S. counterparts at in both cost and quality -- remains to be seen. For the moment, however, the notion of software development as a hard-to-replicate craft is already losing ground. Thanks to the Internet, many open-source projects have plowed through scalability barriers considered insurmountable by 1970s and 1980s-era managers. Many Indian companies now have the freedom to attack other barriers as well. Reducing development to an assembly-line process has failed more times than it has succeeded. Still, whenever it does succeed, managers quickly take notice.

"I think the good news for American companies is that there is this proven thing out there," Douglass says. "When American companies are looking towards outsourcing, whether for lower costs or faster turnaround, they at least have a common vocabulary. American companies that are savvy know that a high maturity model is a pretty good predictor of quality outcomes."

Viewed from this perspective, overseas outsourcing becomes more than a bottom-line issue. It becomes a way to put pressure on those with the least incentive to change: U.S. executives and recalcitrant programmers still holding out for a return to the glory days of the Clinton administration.

"Let's face it. This is a guttural reaction to competition," says one chief operating officer who, like Smith, prefers to keep his name out of the debate. "I think we've created a bit of an artificial economy around what [U.S. developers] believe their market value is. The run-up on compensation in the '90s lasted so long, it created the expectation that people should be paid at a certain level."

For Smith, who would love nothing better than to hire back her U.S. programmers, the dot-com run-up and the current mania for outsourced labor share an eerie similarity. In both cases, she says, top-level managers have valued the actions of competitors and investors higher than the actual information coming back from the marketplace.

"I've talked to a few people in my position," she says. "In general, when senior management makes a decision to outsource, there's political pressure to pretend it's working just so they don't look stupid. That's happening here, too. Everybody has to grin and bear it just so Joe Schmoe at the top doesn't look like an asshole."