

DOI: 10.1145/1400181.1400212

BY AMRIT TIWANA, ASHLEY A. BUSH, HIROSHI TSUJI,
KENICHI YOSHIDA AND AKITO SAKURAI

Myths and Paradoxes in Japanese IT Offshoring

Nothing captures Japanese offshoring practices better than the old Japanese proverb “Tap the stone bridge before crossing.” As global IT offshoring continues to grow at an astounding pace,³ the seemingly impenetrable Japanese market baffles many offshore IT vendors. On one hand, Japanese industries are among the most software-intensive, yet they *appear* reluctant when it comes to offshoring. In this article, we offer six paradoxical insights from field research involving five leading Japanese companies—Toshiba, Fujitsu, IBM-Japan, Hitachi, and Mitsubishi—and their offshore IT vendors spanning China, Vietnam, Philippines, India, Hungary, Romania, Bulgaria, Belarus, Russia, Taiwan, and Sweden to uncover insightful peculiarities in Japanese offshoring practices (see the sidebar “How the Study was Conducted”).

Some of the findings might come as a surprise to managers who are more accustomed to Western—particularly American—offshoring practices (see Figure 1). Overall, our research lends novel insights into how the Japanese have a different interpretation of what constitutes outsourcing, how their offshoring decisions

are driven by knowledge rather than cost or value drivers, their use of circles of trust with a high entry threshold, unusual but surprisingly effective governance strategies, and how they have mastered the sophisticated art of modularization. These paradoxical practices offer invaluable lessons for both Western companies that outsource IT as well as for offshore vendors attempting to gain a foothold in the burgeoning Japanese market.

A Broader Field of View

The widespread misperception that Japanese firms do not extensively outsource IT is rooted in the fundamental differences in meaning of outsourcing in Japanese versus Western interpretations (see Figure 2). The Japanese have outsourced software development for much longer than comparable American firms but their IT offshoring history is brief in comparison.⁴ The most prevalent form of outsourcing in Japan is to direct subsidiaries (i.e., the client fully owns the vendor organization), followed by an increasingly popular affiliate model where the client has a direct financial stake in the vendor. These models are rarely encountered in Western companies, where most outsourcing follows the third and least popular model among Japanese organizations—a third-party vendor bound by a contract.³ The logic is that the first two models minimize conflicts of interests between clients and vendors—or what economists call agency conflicts. In turn, outsourcing relationships are more likely to be a win-win situation for both the client and vendor and contracts more extensive than a handshake are rarely necessary.

The intellectual property-driven and quality-driven nature of the Japanese consumer electronics business—not just cultural norms—create a strong disinclination among these companies to rely on formal or complete contracts (discussed later). The myth that IT outsourcing is less prevalent in Japanese companies holds only for outsourcing to third party offshore vendors (Type C in Figure 2), from where most foreign

vendors' anecdotes about closed doors appear to stem. A widespread corporate practice is to consciously manage the first two—and increasingly all three—types of outsourced projects as a balanced portfolio. Japanese managers are however becoming progressively more open to this model, albeit with some puzzling but surprisingly effective practices described next.

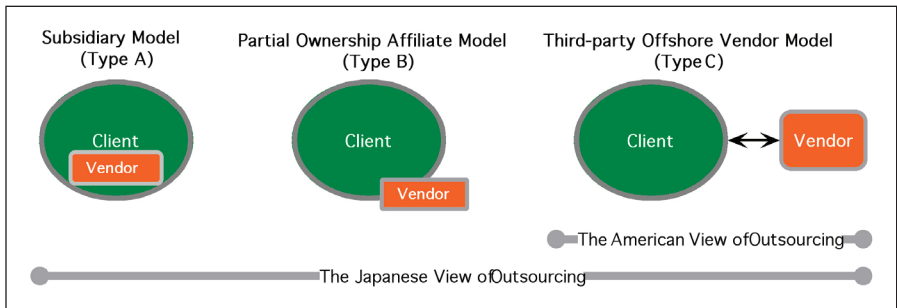
Trust But Verify

We found that IT offshoring decisions of Japanese companies—especially those that involve third-party vendors—are overwhelmingly motivated by access to unique technical expertise rather than by cost-cutting considerations. Japanese managers consistently cautioned us that they only consider offshoring to third-party vendors when the required technical expertise is unavailable in subsidiary or affiliate companies. Western offshoring, in contrast, is motivated primarily by value and cost. This should come as little surprise considering much of the early practices of knowledge management came from corporate Japan.⁸ The prized asset in such rare outsourcing partnerships is the offshore vendor's specialized knowledge. Unfortunately, it is hazardous to trust and almost impossible to verify a vendor's technical expertise upfront.¹¹ Knowledge differences between the client and the offshore vendor therefore expose the client to the risk of being taken advantage of. A peculiar Japanese solution to this dilemma is to invest heavily in verifying the technical expertise of multiple vendors using “dummy projects,” evaluating the process and outcomes, and then blindly trusting

Figure 1: The Vendor Expectation-Reality Gap in Japan

Offshore IT Vendors' Expect...		Japanese Reality
Precise, stable requirements	↔	Volatile requirements
Early involvement in design	↔	Late involvement after design completion
Precise contracts	↔	Informal circles of trust run with handshakes
Global experience as a selling point	↔	Demonstration of technical capability
Domain knowledge acquisition	↔	Domain knowledge protection
Rapid decisions from client-liaisons	↔	Group consensus-driven decisions
Full responsibility for outsourced project	↔	Modular pieces of large projects
Value-add and cost savings motivations	↔	Technical expertise access motivations

Figure 2: Models of Outsourcing in Japanese Organizations



the chosen one on the next project.

One company in our study did this by reverse-generating project requirements for an internally completed project, deliberately introducing ambiguities into the requirements, and giving duplicate development contracts to three different vendors in parallel (see Sidebar “A Tale of Three Vendors”). None of the vendors were aware that the same project was also being developed by the others or that the client already had a completed solution. Internal IT staff evaluated the technical quality of the completed solutions from each of the three vendors using the existing internal software source code from the completed software as a benchmark. In addition to the qual-

ity of the delivered code, the company also assigned weight to whether (and how) each vendor sought clarifications for the intentionally-ambiguous requirements or whether they just assumed them away. The highest ranked vendor—neither the lowest in cost nor the highest in technical quality—made it to the preferred partner list and got the contract for the *actual* project that itself was worth about \$100,000.

Due diligence and vendor screening are much talked about in American corporate circles but this peculiar Japanese practice gives the idea its teeth. Many Western managers unaccustomed to Japanese business practices will undoubtedly scoff at the idea of spending \$100,000 *just to select* the “right” vendor

How The Study Was Conducted

The research was sponsored by the global research and development (R&D) headquarters of Toshiba, IBM-Japan, Fujitsu, Hitachi, and Mitsubishi Electric Corporation and used a multi-case study methodology. We interviewed lead project managers for 15 offshored software development projects in five leading Japanese technology corporations: Toshiba, IBM-Japan, Fujitsu,

Hitachi, and Mitsubishi. We also interviewed 33 other middle-level IT managers who were responsible for offshore project management in these companies. In addition, we also interviewed Country Managers of the Indian and Chinese vendors with which these companies partnered. All interviews were conducted using a semi-structured interview methodology following the case

study methodology guidelines.⁷ Five researchers—which included three native Japanese-speakers—were present for each interview, all of which were conducted in and around Tokyo, Japan. Four interviews were conducted in English and the rest were conducted primarily in Japanese. Each interview lasted between 75 and 175 minutes. All interviews were electronically recorded and supplemented

by written notes taken by each of the five researchers. The recorded interviews and notes were then analyzed using across-case analysis techniques as recommended for multiple-case-based research studies.⁷ The purpose of this analysis was to uncover systematic patterns of IT offshoring practices as well as peculiarities across the cases, which are discussed in this article.

for a \$100,000 project. But offshore vendors must realize that most Japanese managers view this as the beginning of a long-term relationship and not a contracting cost. Japanese companies rely heavily on informal contracting within such *circles of trust*. Neither the most technically adept nor the lowest-cost vendors have any particular advantages in joining those circles.










Small Pieces, Loosely Joined

Many Japanese managers are cautious that offshoring can expose core technologies and business processes to offshore vendors, creating a potential risk of intellectual property theft. This is a legitimate concern, given that the Chinese vendors used by Honda started selling their own replicas of its motorcycles, GM suppliers cloned an entire car, and a Chinese manufacturer replicated Hitachi's 1.8" microdrive used in the original iPod.¹ Since almost the entire software offshoring market consists of Japanese hardware and consumer electronics manufacturers (there are a few pure-play software development companies in Japan), this concern is more the norm than an exception. Theft of intellectual property is notoriously difficult to detect in intangible products such as software compared to manufactured goods. Moreover, since the majority of Japanese offshoring is motivated by knowledge differences between the client and vendor, the likelihood that the vendor has unique technical expertise is high but so is the likelihood that the vendor does not have—but might value—the client's business domain knowledge.

Leading Japanese companies consistently follow a simple but surprisingly effective tactic to proactively mitigate this risk: They decompose an outsourced project into smaller pieces that are outsourced to different vendors in a way that only the client can piece them back together. In technical terms, they protect their proprietary knowledge assets through their mastery of the art of *modularization*.⁹ This approach allows keeping core technologies in-house and also distributes project risk. In contrast, most Western companies attempt to safeguard themselves against this risk through legal contracts, which are frequently unenforceable across national boundaries and in the vendors' countries.

A Tale of Three Vendors

How do Japanese companies select a vendor for projects with high strategic importance? One company in our study has adopted a paradoxical approach for offshore vendor selection that minimizes the risk of failure in projects with high strategic importance. The company offered the same "test" project to three different Chinese vendors to assess their capabilities on several dimensions (none of the vendors were aware that other vendors were simultaneously working on the project). The project had already been successfully completed internally. The test project required the vendors to "port" an existing program to a different platform and language to speed its responsiveness. The company purposely introduced ambiguity in the requirements to see how each vendor would handle it. The following table describes the performance of each vendor on the task.

	Vendor		
	A	B	C
Expected Man months	11	8	16
Estimated cost	¥2.75 Million	¥1.7 Million	¥5.8 Million
Communication	 Fair – Asked questions by email and waited for a clarification before proceeding with further development	 Good – Called for clarifications.	 Poor – Used an interpreter that did not understand technical terms. Gussed specifications.
Delivered quality Current application response speed 50 seconds. The Japanese client requested 24 seconds for new application.	 3.32 seconds	 4.4 seconds	 45 seconds
Delivery (target: 1.5 mo)	1.5 months	1.5 months	1.7 months
Final Assessment	 Good – Vendor understood the intended specifications accurately without imposing an excessive overhead on the client company; the delivered software had the fastest response time.	 Fair – Vendor improved response time but required frequent communication to clarify requirements.	 Poor – Application over specified due to vendor misinterpretations.

In the final assessment, the Japanese client chose Vendor A for their next project. Vendor A, neither the cheapest nor the most expensive of the three, was selected because, figuratively speaking, the vendor was able to get on the same page with the least overhead on the client and then delivered the best implementation (judged as the one with the shortest response time).

Control through Design

We also found that Japanese companies exercise control over the vendor by completely controlling the front-end design rather than using the Western-style strategy of attempting to control the development process or methodology used by the vendor. This approach requires the complete conceptual and architectural design of the outsourced project to be done in-house by the client, which significantly increases the cost of offshoring. In contrast, vendor involvement in the early stages is more common outside Japan. Figure 3 illustrates how the hand-off to the vendor typically occurs much later in the Japanese approach than in the typical American model.

One motivation for doing this in the Japanese companies that we studied stems from the type of software that is outsourced. Unlike the majority of Western companies where the largest chunk of offshoring is for business software applications, these Japanese companies primarily outsourced embedded software (e.g., software code that functions inside a piece of hardware such as a cell phone, laser printer, GPS car navigation system, or a digital camera). The success of such hardware is inextricably tied to the quality and reliability of such embedded software. The second motivation is the strong desire to protect proprietary knowledge by keeping business domain knowledge separate from technical knowledge. Controlling the design and

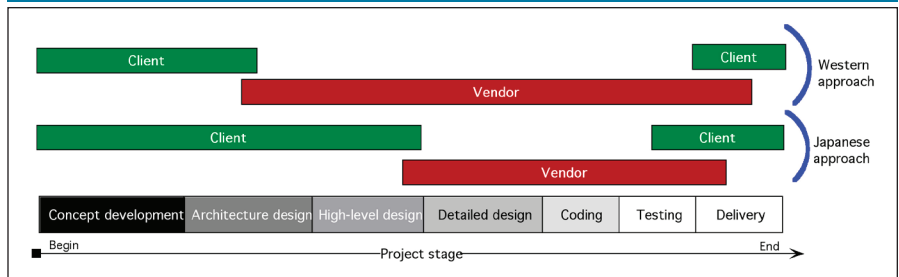
core technology allows Japanese companies to effectively accomplish this without compromising the user interface or functionality of the outsourced project. This also drives the use of “Bridge Software Engineers” discussed later. Third, this strategy allows Japanese engineers to overcome language barriers because the hand-offs to the vendor are language-independent standardized design notations (such as Unified modeling language (UML) diagrams, user interface mock-ups, and flow charts).

The added investment in control through in-house design pays off in the form of a more on-target end result and a lowered need to monitor how the vendor internally manages the development process. Much like their American competitors, extensive use of clearly specified outcome metrics (which are rigorously evaluated by Japanese engineers during acceptance tests in the testing and delivery phases of the outsourced project) is widespread.

Mastering the Art of Synchronized Swimming

The concept of requirements “signoff”—widely used in Western-style offshoring—is a rare practice in Japanese offshoring partnerships. Japanese companies expect requirements to be volatile and constantly evolve over the lifecycle of an outsourced project. A leading, \$2 Billion Indian IT vendor in our study complained that this process difference was the singly most challenging aspect of working with Japanese clients. Since the majority of projects are embedded software or middleware applications—rather than business applications that are more commonly outsourced outside Japan—the performance of the electronics hardware in which the software is embedded is inextricably tied to the quality of this software. Japanese companies have also been very successful in their use of concurrent engineering, where the hardware and software are developed in parallel to shorten time-to-market.⁸ We observed a strong reluctance to discard this well-honed practice. Therefore, design changes during the development of the hardware often trigger necessary changes in the design of the accompanying software, generating volatility in the requirements of the outsourced software components. This requires a vendor to synchronize its in-

Figure 3: Japanese companies exercise project control by completely controlling the front-end design stage.



ternal development process with the other pieces of the total system that are being concurrently developed inside the Japanese client organization. Freezing of requirements at the outset of a project is rarely feasible. The most successful foreign vendors so far have been ones that have come to accept requirements volatility as a given rather than trying to fight it.

Bridges, Not Analysts

Another striking difference in Japanese offshoring relationships is the use of what is described as a “Bridge Software Engineer” (BSE). The concept of a BSE does not have a precise equivalent in most Western organizations, although it resembles a systems analyst or client liaison on the surface. Unlike systems analysts that play an active role in a project during its front-end phases,⁵ the BSE actively and constantly coordinates *knowledge flows* between the client and vendor groups throughout the entire project lifecycle—a key ingredient in software projects.^{2, 6, 10} A typical BSE is usually assigned full time to no more than two projects. The BSE is a mid-level manager from either the client company or in the case of third-party outsourcing partnerships, drawn from a subsidiary or affiliate company in the vendor’s country. A BSE can reside at the vendor’s site, the company’s site, or split time between the two.

The role of a BSE is to maintain the separation of the company’s proprietary business domain knowledge from the vendor’s technical knowledge while ensuring that the two are effectively integrated into the offshore development process. In the projects in our study, the BSE—often bilingual—was the *primary* channel of communication between the client and vendor. (As an aside, we found that weekly video conferencing is the preferred and most heavily used

mode of communication between client and vendor staff.) Japanese managers firmly believe that constant communication is the key factor determining the fate of an outsourced project. The BSE also ensures that stakeholders in the client organization are in agreement about the progress and direction of the work, consistent with the strong emphasis on group consensus—rather than Western-style individualism—in Japanese companies. A BSE therefore serves as a knowledge broker for the contextual business idiosyncrasies and the associated cultural subtleties of the outsourced project. A BSE also serves as a hedge against project risk in lieu of the precisely spelled out contracts in Western-style offshoring partnerships.

Conclusion: Perspective Matters

Offshore vendors intent on winning business in the Japanese offshoring market must remember that they are not in Kansas anymore. Three idiosyncrasies of the Japanese market are noteworthy. First, unlike value-motivated offshoring by Western companies, Japanese offshoring to outside vendors is primarily for access to expertise that is unavailable in subsidiary or affiliate companies. They will go to surprisingly great lengths to verify vendors’ technical expertise. The vendors that pass their litmus tests are usually governed through informal, trust-based relationships rather than formal Western-style contracts. Second, volatile project requirements are the norm and are rooted in the extensive use of concurrent engineering practices that have historically worked well in Japanese companies. Finally, Japanese companies prefer to completely control the design and core technologies of offshored projects and often deploy a BSE to coordinate knowledge integration between the client and vendor staff.

Japanese offshoring practices also lend three valuable insights for Western companies that offshore. First, capital-based relationships with offshore vendors can trounce some of the oft-encountered agency problems and governance challenges. Second, modularization of projects by breaking them up into pieces offshored to different vendors in a way that they collectively cannot put together can protect intellectual property and distribute risk for some types of projects better than legal contracts. This is less feasible with business applications than it is with embedded software. Third, they should consider bringing more of the high-level design back in-house as an alternative to other types of project controls such as output specifications or mandating specific development processes. 

References

1. Anonymous, Fakes!, *BusinessWeek*, Feb 7 2005, 54.
2. Armour, P.A. A case for a new business model: Is software a product or a medium? *Comm. ACM* 43, 8, (Aug. 2000), 19-22.
3. Aspray, W., Mayadas, F., and Vardi, M. (editors) *Globalization and Offshoring of Software: A Report of the ACM Job Migration Task Force*, ACM, NY; www.acm.org/globalizationreport, 2006.
4. Cusumano, M. The puzzle of Japanese software. *Comm. ACM* 48 (2005), 25-27.
5. DeLone, W., Espinosa, A., Lee, G., and Carmel, E. Bridging global boundaries of Is project success. In *Proceedings of Proceeding of 38th HICSS Conference*, Hawaii, 2005.
6. DeSouza, K. Barriers to effective use of knowledge management systems in software engineering. *Comm. ACM* 46, 1, (Jan. 2003), 99-101.
7. Eisenhardt, K. Building theory from case study research. *Academy of Management Review* 14, 4, (1989), 532-550.
8. Nonaka, I. The knowledge-creating company. *Harvard Business Review*, (Nov/Dec 1991), 96-104.
9. Parnas, D. On the criteria to be used in decomposing systems into modules, *Comm. ACM* 15, 9, (Sept. 1972), 1053-1058.
10. Robillard, P. The role of knowledge in software development, *Comm. ACM* 42, 1, (Jan. 1999), 87-92.
11. Tiwana, A. Beyond the black-box: knowledge overlaps in software outsourcing, *IEEE Software* 21, 5, (2004), 51-58.

This research was sponsored by Hitachi, Fujitsu, Mitsubishi Electric, IBM-Japan, NEC, and Toshiba and funded through Joint Forum for Japan's Strategic Software Research (SSR) and International Information Science Foundation.

Amrit Tiwana (tiwana@iastate.edu) is the Union Pacific Professor at Iowa State University, Ames, IA.

Ashley A. Bush (abush@fsu.edu) is an associate professor at Florida State University in Tallahassee, FL.

Hiroshi Tsuji (tsuji@cs.osakafu-u.ac.jp) was with Hitachi and is a professor at Osaka Prefecture University, Osaka, Japan.

Kenichi Yoshida (yoshida@gssm.otsuka.tsukuba.ac.jp) is a Professor at University of Tsukuba, Tsukuba City, Japan.

Akito Sakurai (sakurai@ae.keio.ac.jp) is a professor at Keio University in Tokyo, Japan.

© 2008 ACM 0001-0782/08/1000 \$5.00