Michael A. Cusumano

The Puzzle of Japanese Software

Searching for answers to the many questions of software production.

During the past two decades, Japan has produced and used an enormous amount of software. Its domestic market is second only to the U.S. in terms of sales. But Japan's software business—the activity of creating software products and services for sale around the world—has been a puzzle to me and other observers of Japan for many years. Excluding billions of dollars worth of "hard" products containing embedded software, which range from machine tools to consumer electronics and automobiles, Japanese firms make relatively few software products. Japan also exports little standalone software, with the exception of video games for Japanese hardware platforms made by Sony and Nintendo.

The worldwide software industry in the past few years generated approximately $600 to $700 billion per year in revenues for software producers, excluding services such as systems management and data entry. About one-third of the software revenues are standardized products and the other two-thirds are services related to software development and implementation, such as system customization and IT consulting. The U.S. and Canada account for about half of the worldwide software business and Europe approximately 30%. Asia accounts for 15% to 20%, with Japan comprising about 10% of the world market and represents the largest single Asian market.

To be more precise, according to Japan's Ministry of Economics, Trade, and Industry, Japan's software industry revenues in 2003 totaled an impressive $73 billion (8081 billion yen). This number includes 82% in custom-built software and merely 18% in standardized or "packaged" products, including software games. Eliminating software games, Japanese packaged software development is merely a few percent of the total industry.

There seem to be various reasons why we see little Japanese software outside Japan and so few software products: Japanese do make many software products, but they are generally written with Japanese interfaces, many are for mainframe computers, and

ROBERT NEUBECKER

they are often heavily tailored to Japanese needs, such as for generating local maps or doing local accounting and taxes. Japanese corporate users have traditionally relied on mainframes and were slow to move to personal computers that used a wide variety of relatively inexpensive products. Japanese corporate IT departments still seem to prefer expensive custom-built software, albeit with increasing amounts of packaged U.S. software products mixed in, such as databases and operating systems.

To take full advantage of their process skills and knowledge of software technology in a wide variety of applications, Japanese firms might try harder to do what they have done in other industries.

Another problem is that Japan's computer producers competed for decades with their own incompatible proprietary standards and were slow to adopt global platforms such as Windows and Unix, making it even more difficult for them to participate in the global software products business. Many Americans and Europeans, and some Japanese, also believe that Japan's education and employment systems suppress the type of creativity needed to cultivate software programmers with a creative product mentality. The fact that we see Japanese creativity in video games and consumer electronics more broadly makes me wonder about this last generalization, but it may be true nonetheless.

Like the U.S. and Europe, Japan has a long history of software development, dating back to the 1960s. The major computer manufacturers—Fujitsu, Hitachi, and NEC, and their software-dedicated subsidiaries—rank among the largest software producers in the world, with thousands of employees and billions of dollars in software revenues. NTT Data and CSK are Japan's other major software producers. But these and most other Japanese software companies, with the exception of video-game vendors, are largely custom shops and system integrators. It is difficult for them to develop domestic and global economies of scale with these types of businesses.

Whatever the reasons, Fujitsu, Hitachi, NEC, and their specialized software subsidiaries, as well as other large software companies such as CSK, have rarely tried to develop global software products. To put Japan's experience in context, however, we should also recognize that most European countries, as well as India and China, also produce far greater amounts of custom software than they do packaged software. Europe consists of many small fragmented markets. India is focused on software as a service. And the Chinese software market, though it has great potential to develop its own standards and products, is still in its infancy. Overall, comparisons with other regions and countries suggest that the most unusual software business is found in the U.S., not the rest of the world.

Be this as it may, since the 1960s Japan has used its large domestic market as a platform for developing global products in a variety of industries. For this reason alone, its lack of effort or success in software remains puzzling. I have been studying Japanese management practices since the late 1970s. I looked first at production and engineering management in the automobile industry, and then consumer electronics, and then the software industry from around 1985. Some readers may be familiar with my 1991 book on Japanese software factories [1], which documented the efforts of Hitachi, Toshiba, Fujitsu, and NEC to structure large-scale software development, particularly semi-customization projects for industrial customers, with an intense focus on project management, reuse promotion, and quality assurance activities.

To attempt to ascertain the differences between Japanese and U.S. software more quantitatively, I did a study in 1990 with Chris Kemerer (then of MIT and currently with the University of Pittsburgh) that compared U.S. and Japanese software projects from major firms. Our sample was small but we found that Japanese projects had one-fourth the number of defects reported by customers in the first 12 months after shipment compared to U.S. projects (median of 0.2 defects for 11 Japanese projects versus 0.8 for 20 U.S. projects) [2].

In a 2003 study, conducted with Kemerer, Alan MacCormack of Harvard Business School, and Bill Crandall of Hewlett-Packard, based on a global sample of over 100 projects, we found even more remarkable levels of quality in Japan, (again, measured as defects reported by customers in the first 12 months after shipment). The 27 Japanese projects reported one-twentieth the numbers in 31 U.S. projects—a median level of 0.02 defects per 1000 source lines of code, compared to 0.40 for the U.S. sample [3]. The Japanese numbers are so strikingly good that we have decided to investigate the projects more closely and hope to report on our analysis in a future article.

Clearly, Japanese companies have been able to apply their experience in quality control, engineering management, and testing techniques for other products to software development. It is difficult to talk about causality here, but so few bugs may suggest an overly rigid style of development and a preoccupation with "zero defects" rather than innovation and experimentation, at least at the firms we sampled. Creating bugs is a cultural "no-no" in Japan, but innovative software always requires some trial and error.

It is also surprising that Japanese programmers not only write the fewest bugs, but they continue to be highly productive in terms of the amount of code they write. The best Japanese companies, in other words, seem to be very good in the process of generating or producing software. Perhaps this is the legacy of the "software factory," which was once so popular in Japan. Japan's greater problem, however, seems to be the products that these organizations create—or don't create.

To take full advantage of their process skills and knowledge of software technology in a wide variety of applications, Japanese firms might try harder to do what they have done in other industries. Like Microsoft and many U.S. companies, and some European companies (such as SAP and Business Objects), they might try harder to identify horizontal market needs, package functionality more effectively, and run their software companies and IT divisions more like scale- and efficiency-oriented businesses.

As I have written elsewhere, and will report on more in the future, the global software products business has its own very serious problems with falling prices, piracy, and very good free, open source software. These trends represent the maturity and commoditization of the traditional software products business, and the shift in software revenues toward services and maintenance from an installed base of users. But these broad trends toward commoditization make it all the more important for software producers in Japan and elsewhere to take advantage of whatever process skills they have and put their abilities to work producing software—both standardized products and custom-built systems—as efficiently as possible. C

References

1. Cusumano, M. *Japan's Software Factories*. Oxford University Press, NY, 1991.
2. Cusumano, M. and Kemerer, C. A quantitative analysis of U.S. and Japanese practice and performance in software development. *Management Science 36*, 11 (Nov. 1990).
3. Cusumano, M., MacCormack, A., Kemerer, C., and Crandall, B. Software development worldwide: The state of the practice. *IEEE Software 20*, 6 (Nov.–Dec. 2003), 28–34.

Michael Cusumano (cusumano@mit.edu) is a professor at the MIT Sloan School of Management and author of *The Business of Software*, Free Press, 2004.

This column is adapted from the Japanese introduction to M. Cusumano, *The Business of Software,* Free Press, NY, 2004; translated as "Sofutouea kigyo no kyoso senryaku" (Tokyo, Daimond, 2005).