



CENTER ON JAPANESE ECONOMY AND BUSINESS

日本経済経営研究所

Working Paper Series

July 2013, No. 324

Academic Entrepreneurship: A Comparison of U.S. and Japanese Promotion of Information Technology & Computer Science

Robert Cole

This paper is available online at www.gsb.columbia.edu/cjeb/research

C O L U M B I A U N I V E R S I T Y I N T H E C I T Y O F N E W Y O R K

Academic Entrepreneurship: A Comparison of U.S. and Japanese Promotion of Information Technology & Computer Science*

Prepared for Stanford Project on Japanese Entrepreneurship.
Entrepreneurial Policy, Outcomes, and Strategies in Japan:
Lessons for the Rest of the World. June 6-7, 2013

Robert E. Cole

Joint Researcher, ITEC, Doshisha University

Emeritus Professor, UC Berkeley, Haas School of Business

*The author gratefully acknowledges the funding of this research by grants from: the Institute for Technology, Enterprise and Competitiveness, Doshisha University, Kyoto, Japan; the California Management Review; the Garwood Center for Corporate Innovation, Institute for Business Innovation at the Haas School of Business, UC Berkeley. I am indebted to the feedback, help, and advice I received from the following individuals: Fujimoto, Takahiro, Shinya Fushimi, Kenji Hiranabe, David Hodges, David Mowery, Yoshifumi Nakata, Katsutoshi Shintani, Tetsuo Tamai, Hirofumi Tatsumoto and Masato Takeichi.

Abstract

Manufacturing hardware is increasingly a commodity product with low margins. The reason for the diminished value of hardware is that more and more of the value in high-tech products is contributed by software meeting user needs, through enabling new functionalities and services. Those firms better able to create and use software advances, improve their competitive outcomes. Japanese high-tech firms have been slow to recognize and act on the growing importance of software and have suffered competitively. The reasons are many. It is widely recognized that human capital is a critical component of software innovation and thus would be central to any explanation of Japanese firms' weakness in software. For this reason, I focus on the role of university engineering education in IT, comparing the U.S. and Japan. My analysis documents the leadership role played by U.S. universities and in particular, the academic entrepreneurship demonstrated by leading computer science departments. The contrasts with leading Japanese universities are striking. On the Japanese side, they include a slowness in recognizing the importance of software and in adopting state of the art curriculum, a failure of MEXT to regulate the way in which universities implemented their mandate to develop information technology, an egregious sabotaging of the new information technology departments by university administrators, and a reluctance of leading firms to hire computer science graduates. Finally, centralized faculty decision making allowed engineering faculty in other departments to resist changing student quotas (teiin) in favor of the new discipline. Taken together, these factors inhibited the development of computer science as a distinctive discipline and put a break on any faculty entrepreneurs seeking to promote the new discipline. By contrast, I will show how institutional practices in the U.S. acted to promote academic entrepreneurship enhancing the growth of the new discipline.

Academic Entrepreneurship: A Comparison of U.S. and Japanese Promotion of Information Technology & Computer Science

How do we explain why the world's third largest economy, Japan, second only to the U.S. in software sales, nevertheless, lags so significantly in other key measures of software performance? One can measure these lags in its miniscule software and information services exports, large software and information services imports, absence of path breaking software firms, absence of a large independent software sector, and its growing weakness in patentable software relative to the U.S. In this paper, I aim to add a piece to the puzzle solution by examining U.S. and Japanese university efforts to provide a new generation of students with the knowledge, analytic skills and tools to address the rapidly evolving field of information technology and computer science (CS).

I begin by documenting one of the key challenges for the Japanese software industry, software innovation. To this end, I examine results from Arora, Branstetter and Drev's recent study of US and Japanese IT patents granted by the U.S. Patent Office (USPTO) from 1983-2004.¹ The authors examined U.S. and Japanese embedded, application, system and enterprise application software patents in the high tech sector. They found that IT innovation has become increasingly software intensive. Researchers first looked at hardware patents in the high tech sector from 1990-2003. They found that these hardware patents increasingly cite software patents. This is true in patents emanating from all countries including Japan. How big a change has there been? For the total dataset, an IT patent granted in 2003 was almost 3.2 times more likely to cite a software patent than one granted in 1990. This represents a remarkable shift over just 13 years suggesting the rapidly growing importance of software.²

Even though this trend is apparent in both Japanese and U.S. patents, the researchers found that Japanese IT hardware patents were systematically less likely to cite software patents over time than non-Japanese firms. Consistent with this finding, they found that Japanese firms file fewer software patents than their U.S. counterparts and that the difference has grown steadily since the late 1980s and especially after the mid-90s. Their findings indicate first that IT inventions have become much more software intensive over time. U.S. firms also have more actively incorporated software inventions into their products and services than have their Japanese counterparts. Even more telling, they find that U.S. firms have improved their innovation performance vis a vis Japanese firms in just those IT segments which are most software intensive (computers and peripherals). The researchers also find that within IT, Japanese firms are now disproportionately located in less software intensive sectors.³

Innovation performance was measured in a number of different ways, including patent productivity per dollar invested in R&D. Using these and other measures, they found that the

software innovation performance of Japanese IT firms increasingly lags behind that of its U.S. counterparts, particularly in IT sectors which are more software intensive. They conclude that their findings are consistent with, and provide a new explanation for the “precipitous global decline” of Japan’s IT industry in the 1990s.⁴

Sources of the Innovation Gap

The research reported above is important. It leads to a second question which is the primary focus of our analysis. The authors discuss two possible sources for Japan’s shortfall in software innovation. First they point to a constrained supply of software knowledge and skills (human capital) in Japan and second to the slowness of Japanese IT firms to recognize the transformational nature of software. They set up a test to differentiate between the two. They reasoned that if Japanese firms are constrained by their software human resources at home, then they will have the incentive to tap into foreign knowledge and expertise by setting up software intensive R&D facilities abroad (e.g., Silicon Valley) to further software innovation. The researchers further reasoned if the Japanese deficit in innovation performance was because the Japanese managers downplay the importance of software, then the research output of their overseas subsidiaries, like that of their domestic operations, ought also to be less software intensive than their American counterparts. Using the same dataset, they find support for their first hypothesis that Japanese firms, constrained domestically by their limited software human resources, further their software innovation in their U.S. subsidiaries; to wit, the share of software patents of total patents invented and patented in the U.S. by U.S. based Japanese firms is much higher than the share of software patents of total patents invented in Japan and patented in the U.S. by Japanese parent firms.

There are, however, problems with this test. The authors assume homogeneity in the patents Japanese file in the U.S. and Japan by relying on filings with the USPTO to measure both U.S. and Japanese software inventive activities.⁵ Yet, it is more expensive to file in foreign jurisdictions, reflecting the costs of translation, retaining offshore legal talent, etc. Moreover, while standards for review, appeal of reviews, and issue of patents have formally been harmonized across high-income economies, differences in all of these persist, further raising the costs and uncertainty of offshore patent prosecution.

For all of these reasons, those conducting empirical analyses of patent data rarely assume that “home-base” and “offshore” patent applications are homogeneous. In line with this thinking, it would be wise to control narrowly for product characteristics. There appears to be a “home country” bias in that U.S. inventors have a home court advantage so that they file more patents in the USPTO compared to foreign applicants for a given set of inventions.⁶ One might expect, for example, that Japanese firms likely select the IT patents they choose to develop in their U.S. subsidiaries and file, from among those with the most promising export/sales

potential to the U.S and other global markets. This, not implausible scenario, may lead to the outcomes they found (Japan's innovation shortfall), an outcome which may have very little to do with software human resource deficits in Japan. With this critique in mind, it is wise to look elsewhere for clues as to the source of the relative weakness in Japanese firm's software innovation. It is in this context that I turn to an examination of university IT education.

University IT Education:

Given the criticality of human capital to software development, we can partially evaluate a nation's recognition of the importance of software in terms of its timing and quality of investment in software education.⁷ Steinmuller argues that the U.S. software industry's advantage over Japan lay in its first mover advantage, one that was enabled by government R&D policy and the early development of CS education at the university level.⁸ The latter advantage over Japan and many other nations, as measured in terms of the quality of U.S. CS education, persists to this day.

An examination of the formal education of software engineers in Japan and the U.S. reveals somewhat similar distributions when comparing high school, junior college, and university graduates. For example, 53% of Japanese software engineers (2007) are reported to be University graduates vs. 48% in the U.S. (2008).⁹ The problem with these data is that they don't reveal what percentage of these graduates actually had any kind of formal IT education. It is a widely-held view among leading Japanese software specialists that well over half of software engineers have had no formal education in the field but acquired their skills through company training. This is particularly likely to be the case in the lower tiered IT service firms building customized software.¹⁰ Unfortunately, I have been unable to find comprehensive data documenting these common views. The number working in software in the U.S with no formal education in software has also been large in the past. Over time, however, those with formal software education have increased. As we shall see below, those with undergraduate CS degrees have increased rapidly in recent years in the U.S. but much less so in Japan. Analysis of NSF data show that as of 2003, 85% of U.S. computer software engineers had at least one Science and Engineering degree, defined broadly as those with degrees in computer/math sciences, biological, environmental, and life sciences, physical sciences and social science and engineering. Of these, 41% had at least one degree in computer/information sciences and those whose highest degree was in computer/information sciences constituted 38%.¹¹

It is especially revealing that some 20% of software engineers in the U.S. have a graduate school degree compared to 10% in Japan.¹² The gap in PhDs is even larger than suggested by this 10% difference. Until recently, the majority of Japanese PhDs in engineering were earned by industrial researchers in Japanese companies. These degrees (ronbun hakase) are awarded by the employees' former university, typically after many years of R&D research. No

coursework is necessary, only submission of a dissertation with some articles published in well-regarded journals. These PhD degree holders are very unlikely to be on the forefront of software innovation. These type of PhDs are in sharp decline. Putting aside the ronbun hakase degrees, most CS (and most Japanese engineering PhDs for that matter) are earned by those aiming for an academic career. By contrast, most CS PhDs from U.S. universities are hired by industry or government.¹³ A great deal of new value in the private sector has been and is being created in the U.S. by PhD and Masters of Science students and engineers. Some 22,000 PhDs have been granted in CS and IT in the United States between 1978 and 2008.¹⁴ PhD students at the best U.S. universities experience rigorous coursework and practicums and not a few go on to envision and then create and sustain a whole new world, e.g. UNIX, relational databases. Startups founded by those with PhDs include Adobe, Qualcomm, Google, Sun, Cadence, Synopsys, VMware, and Symantec etc. Still other PhDs like Edgar Codd, with a PhD in CS from the University of Michigan, worked at IBM but nevertheless inspired startup firms like Oracle to develop commercial products building on his path breaking research on relational databases. This is where the Americans have excelled.

One is hard put to compile a comparable list of path breaking equivalents in Japan. It is not uncommon at major U.S. IT companies to see CTO's with PhDs. Moreover, a comparable list could easily be compiled for graduates with Masters of Science degrees. To be sure, the Japanese system is changing: by the period 2007-2009, the dissertation only degrees declined to some 27 % of all information science and software dissertations but the pace of change is slow and the benefits of the new distribution are yet to be revealed. Behind the limited numbers of Japanese PhDs in software related private sector jobs is the long term lack of demand by Japanese industry for engineering PhDs overall and the paucity of software startups.

There is further evidence of Japan's lagging status from the 2012 Academic Rankings of World Universities conducted at Shanghai Jiao Tong University. In 2012, some 430 universities reported their data on students, academic staff and resources. The participating universities cover 61% of the Top 100 universities and 50% of other Top 500 universities on the Academic Ranking of World Universities 2012. The data were augmented by data for 1,200 universities, including alumni and staff winning awards and medals, highly cited researchers, and papers in leading journals. The rankings for CS show that U.S. universities captured 20 of the top 24 rankings including the top 9. The only Japanese university in the top 200 universities in CS was the University of Tokyo which ranked between 150-200.¹⁵ The foregoing rankings are hardly definitive. In part, they capture globalization of programs which is not equivalent to excellence per se. Weak English language capabilities and limited numbers of Japanese students studying abroad hold down the Japanese rankings. Nevertheless, the rankings do suggest that Japan is far from the forefront of CS and information technology education.

Just as telling has been the long slow process of incorporating state of the art software knowledge into the curriculum of Japanese Faculties of Science and Engineering. It was well recognized by CS professors in Japan in the 1990s that CS faculties in Japanese universities were weak.¹⁶ They relied heavily on U.S. created standards of IEEE-CS (Institute of Electrical and Electronics Engineers-Computer Science) and ACM (Association of Computing Machinery), but with long delays.

The 1997 (J 97) standards published by the Information Processing Society of Japan (IPSJ) were greatly influenced by the U.S. "Computing Curricula 1991" of IEEE-CS and ACM.¹⁷ Though these curriculum were already outdated by the time they were adopted in the U.S., not to speak of when they were adopted in Japan in 1997, they nevertheless lifted the modest profile of software in Japanese IT curriculums.

Still later, a completely revised set of standards was completed in 2007 (J07); they were heavily influenced by the CC2001 of IEEE-CS and ACM. The timing of both Japan curriculum revisions reveals a significant lag from the initial American revisions. To be sure, the J07 standards, when published, incorporated CC2005, but the fact stands that the Japanese educational establishment has been a slow follower when adopting software curriculum.¹⁸ Given the rapidly changing hardware and software technology during these periods, this lag suggests a lack of appreciation by the Japanese educational establishment of the speed at which the technology was changing and the importance of quickly accommodating to these changes. Along similar lines, the IPSJ's accreditation committee criticized Japanese CS and other related curriculums for giving priority to desk study while the American university curriculum emphasized software design (exposing students to experiments and practicums).¹⁹

Critical to the incorporation and communication to students of up-to-date curriculum is faculty with up-to-date knowledge. Here, Japanese universities were at a disadvantage. Much new IT knowledge was being developed in the U.S, especially with the advent of the PC era in the early 1980s. At the same time, Japanese firms which had achieved significant competitive success through modeling IBM's mainframe architecture, remained committed to building new generations of still larger mainframe computers.²⁰ This misplaced focus of the incumbents put them still further behind the U.S. in developing, integrating and deploying new software knowledge.

The expansion of IT departments in Japanese universities in the 1980s was associated with a shortage of CS-trained faculty. To fill positions, many universities hired "retired" IT executives from major companies, like NEC, Fujitsu, Hitachi and Toshiba. It was and still is common for the Ministry of Education to push new departments in applied fields like technology and business to hire a portion of their faculty from among those with practical experience. These new academics helped fill out the faculty staff IT departments in the 1980s and 1990s. While they

had practical experience in IT business issues and some technical matters, most of their corporate training and experience was in mainframe hardware and software. In the U.S. these were quickly being displaced by PCs, workstations, newer programming languages, networked systems, the Internet, and the like.

Most of the Japanese faculty with these corporate backgrounds, were not competent to teach university students, much less the state of the art or the state of best practice. To be sure, a few made important contributions. Such was the case of Nobumasa Takahashi, who came from Hitachi but his contributions were more in the way of administrative field building initiatives rather than research contributions. Most of those with corporate backgrounds were not capable of making contributions on the research front where research scientists, mostly from the U.S., were transforming the field of CS. All this contributed further to the Japanese falling further behind the US and Europe in software during these formative years.²¹

The Education Ministry (MEXT) nominally plays a strong role in the development of university curriculum for both private and public universities. Typically, they approve the plans for the curriculum of a new department and then monitor performance for some four years until the new students' cohort graduate. In practice, the Ministry has relatively little content knowledge (despite having faculty committee advising the Ministry on proposals for new schools and departments and monitoring new schools and departments). It left devising content to the university department's discretion. The Ministry's monitoring role consists of insuring that the new department executes the curriculum plan according to what they promised they would do in their original proposal. In short, they perform typical bureaucratic scrutiny which insures consistency but limits flexibility and adaptation. Accordingly, while they were involved in the development of the 1997 standards, they were absent from the development of the new 2007 software curriculum.

Professor Nobumasa Takahashi, a key player in the activities of the Information Processing Society of Japan, analyzed the national universities which typically set the direction on new technical curriculum. He observed that the Japanese postwar university departmental structures were shaped by continual expansion in the fields of civil engineering, machinery, EE and chemistry. In many cases, new departments were formed from the Ministry of Education's (MEXT's predecessor) budgetary appropriations demanded by these existing departments. "As a result, the new departments were created with the strong coloration of colonies of the old departments."²² In 1998 at a university/industry meeting, industry people complained that the old disciplines were still casting a long shadow over the new disciplines of CS, software engineering and information systems.²³ The clear implication here is that entrepreneurial initiatives by the new software oriented departments were constrained by the old established engineering departments.

Other constraints on the new departments of informatics (roughly equivalent to what Americans call information science) and CS arose from an unanticipated interaction of two developments. First, in 1991, a de facto Ministry of Education deregulation of universities took place which no longer required science and engineering schools as well as other schools to offer liberal arts for the first two years of undergraduate education. Thus, many of the science and engineering schools substituted engineering and science courses in their stead. As a consequence, a large number of redundant faculty were created. Since they could not be fired, they needed to be relocated from their original departments.

Second, at roughly the same time, the Ministry, concerned about declining national birth rates, imposed limits on the expansion of university departments and faculties. They made a few exceptions, however, for growing fields of national importance, one of which was for information technology. The universities took advantage of this exception to “offload” some of their redundant faculty to these growing fields. Many of the transferred faculty continued to teach their existing specialties despite now being in CS departments.

The net result is that in 1998, Prof. Masato Takeichi found that the total faculty (2,615) of the Japanese information science departments in Schools of Science and Engineering in 1998 were constituted roughly as follows: 30% were core CS faculty whose academic specialties were in CS and mathematical informatics. Viewing information science more broadly to include system informatics, electrical informatics and intelligent informatics added another 45%. A remarkable 25%, however, had educational specializations with little or nothing to do with CS.²⁴ The Ministry did nothing to combat this blatant misuse of the filling of the new faculty slots. Ideally, the new dynamic fields of informatics and CS were expected to make strong contributions to software education and to the economy as a whole. Instead, the Ministry actions and then inactions, at a formative stage in development of these new fields, led to the new departments being hamstrung and weighed down by excess, and in some 25% of the positions, irrelevant faculty baggage. In turn, this meant that students had far less access to up-to-date knowledge in these rapidly evolving core fields. More subtly, it made it more difficult for CS to be recognized as a distinctive discipline.²⁵

The latter problem continues to this day. Prof. Fumihiko Kimura, a member of the Science Council of Japan, at a public forum held at the University of Tokyo in March 2013, stated that CS (called *johoukagaku* in Engineering Schools) is still not recognized by most interested parties in Japan as a clearly defined discipline.²⁶ One measure of a new field’s coalescence into a recognized discipline is the willingness of those in established disciplines to contribute to the emerging discipline. This both reflects existing assessments of the emerging discipline and bestows further legitimacy on the new field. In the case of the U.S., the field is closely intertwined with other disciplines. For example, both theoretical and applied mathematicians

played a key role in building the new CS discipline. Mathematical logic, the theorems of Turing and Godel, Boolean algebra for circuit design, and algorithms for solving equations and other classes of problems in mathematics played strong roles in the early development of the CS as a discipline.²⁷

Indicative of the early recognition of CS in the U.S. was the formation of CS departments and sometimes equal partnership with electrical engineering (EE&CS) departments as at MIT and UC Berkeley. Stanford and UC Berkeley established their respective current CS departments and partnerships with EE in 1965. MIT recognized the growing CS activity within its EE Department and in 1975 changed its name to EECS. However, the Laboratory for CS (LCS) was founded in 1963. Originally known as Project MAC (Multiple Access Computing and Machine-Aided Cognition), its mission was to develop a computer system accessible to a large number of people, and to exploit the computer as an aid to research and education. LCS members and alumni have since been important contributors to the development of the ARPANET, the Internet, Ethernet, the World Wide Web, time-shared computers, RSA encryption and dozens of other technologies.

The contrast with Japan is striking. At the University of Tokyo, The Information Engineering Department was established in the Graduate School of Engineering was established in 1972 but the Department of Information and Communication Engineering for undergraduates was not established until 1991.²⁸ The Department of Information Science in the Faculty of Science (most comparable to CS departments in the U.S.) started in 1975 as an augmented successor to the Information Science Laboratory, and has since offered undergraduate and graduate degree programs. Prof. Masato Takeichi, Dean of the Graduate School of Information Science and Technology at the University of Tokyo from 2004-2007, states that mathematicians have not been as drawn to the new IT fields in Japan as was the case in the U.S. He attributes this to CS not being seen a distinct discipline in Japan. Contributing to this weakness, Japanese CS researchers tend to focus on applied rather than basic research.²⁹

CS still has relatively low status in the hierarchy of engineering related fields in Japan and their graduating students are not in high demand in industry. The most common explanation given for the reluctance of Japanese firms to hire CS graduates, is that they do not have sufficient "domain knowledge".³⁰ Given the history of Japanese manufacturing firms in providing domain specific training, it seems an odd explanation. Still another problem reported by knowledgeable software specialists is that the quality and quantity of Japanese high school students choosing to study software fields at the university level is declining. Students appear less interested in IT than in the past and the reason appears to arise from the structure of the

industry with its heavy domestic outsourcing of customized software development and limited opportunities for employees to develop and capitalize on specialized skills.³¹

This description greatly contrasts with evolving trends in the U.S. CS enrollments hit their peak in the year 2000 and then suffered a sharp decline after the dot.com collapse in 2001-2002, with the decline continuing through 2007. Since then, however, CS has exploded in popularity with its graduates increasingly in high demand.³² National undergraduate CS enrollment has grown sharply for five years in a row through 2012.³³ This growth is driven by perceived entrepreneurial opportunities and also by growth in overall corporate demand for the services of graduates of such programs. Overall, some 63,300 U.S. CS and mathematics majors received bachelor degrees in 2009 compared to some 16,300 in Japan.³⁴

The acceleration of CS enrollment growth in the U.S. can be seen at Stanford University, one of the U.S.'s premier engineering schools. Those undergraduates graduating in EE total 50 a year and the number has been stable for some time but the number of those graduating with CS degrees has been on a sharp upward trajectory in recent years. In 2012, some 250 students graduated with CS majors, five times more than EE majors. Even more telling, CS is now the largest undergraduate major at Stanford across the whole university.³⁵ The same trend is evident at UC Berkeley. CS in the College of Letters and Science is the second fastest-growing large major. Its numbers more than doubled, from 138 to 357 (159 percent) from 2010 to 2013.³⁶ Moreover, its combined EE-CS Department in the Engineering College projects that its undergraduate upper division courses will have a ratio of roughly 3:1 in favor of CS for 2013-2014.³⁷

At the leading U.S. engineering schools, those departments like CS, which experience sharp growth, drawing on a highly qualified applicant pool, have the discretion and strong incentives to grow their department. Faculty can be motivated to "put their brand on" a new generation of smart young people and they also use growth to claim greater departmental resources. The incentives for faculty growth are on display at MIT. In their combined EE-CS department in early 2012, the percentage split between EE and CS faculty was 54% to 46% in favor of EE. In Spring 2013, the combined faculty of this department voted to hire faculty at a ratio of 2:1 CS:EE for the next 3-4 years to address the shift toward CS in undergraduate enrollments while balancing this shift with the still greater number of graduate EE than CS students.³⁸ Central administrators typically are pleased with such growth and as seen in the MIT case, even in combined departments such as at MIT and Berkeley, there is seldom any successful effort by non-CS faculty to hinder adjustments to CS's growing student numbers. The historically decentralized education system in the United States means that the Department of Education, unlike its counterpart in Japan, has been a non-actor on such matters. As expressed by David Hodges, former Dean of Engineering at UC Berkeley, "deference tends to be given to

those faculty and departments which show leadership.”³⁹ This captures the entrepreneurial spirit which underlies such rapid adaptation to new circumstances.

The contrast with Japanese universities is stark. We may use the University of Tokyo, Japan’s leading information and communications technology department, as case in point. In the engineering faculty, each undergraduate major, like information and communication engineering (not entirely equivalent to CS at an American university)), has a prescribed student quota. Until the University deregulation of 2006, these quotas were rigidly prescribed by MEXT. Since the 2006 deregulation, enrollment limits became the responsibility of faculty. The faculty in question, however, is not the faculty of specific engineering departments but rather the entire engineering faculty. Thus when discussion of allocation occurs, each department tries to protect its turf and not suffer losses at the expense of other departments. This makes it difficult to reach a consensus on changing the distribution of allotted student quotas. The quota for the departments within the Faculty of Engineering at the University of Tokyo has been changed only once since the enrollment limits became the responsibility of the faculty in 2006. In 2008, they changed in conjunction with a restructuring and consolidation of the various departments. Because of the restructuring it is hard to say how the enrollment of information and communications technology fared. The national universities, having been so long dependent on MEXT (Ministry of Education, Culture, Sports, Science & Technology), find it hard to exert their independent will. They often continue to informally ask MEXT officials for advice before making changes.⁴⁰ Solving the problem by expanding the pie through increasing overall enrollments is a hard case to make in view of Japan’s declining college population and inertial forces.

In the case of the University of Tokyo’s Information and Communications Engineering Department the admission quota as of May 2012 for electrical and electronics engineers (a combined total for juniors and seniors) was 150 while it was only 80 for information and communications technology.⁴¹ This disparity persisted even after the adjustments of 2008 noted above.

In other words, the allocation of student quotas between EE and information and communications technology appears more suitable for two decades ago when IT and software, in particular, were not such global competitive forces. Instead, the current distribution supports traditional manufacturing (monozukuri) education and again shows how the established disciplines continue to constrain the field of information technology. To be sure, the EE education which Japanese students receive today, as is the case in the U.S., incorporates far more CS into their coursework than was the case in the past. Anecdotal evidence from UC Berkeley and the University of Tokyo, however, reveals a much stronger dose of required software courses in the UC Berkeley EE curriculum than is the case with the University of Tokyo.

Finally, Japanese universities find themselves in a bind in that even if farsighted faculty believe in the need to rapidly shift toward CS and were able to overcome resistance from other departments, industry shows no great enthusiasm for hiring large numbers of their graduates at this time. To be sure, one must be careful in making too sweeping generalizations here since CS related programs and curriculum appears in multiple schools and departments in many Japanese universities including the University of Tokyo. This is also the case in the U.S. Further research on this matter is called for.

In summary, there is ample evidence that the Japanese educational establishment was slow to incorporate strong software education into the curriculum and that it still lags. They also failed to insure an up-to-date curriculum was developed and executed by a large and capable faculty. Arguably, this situation contributed to human resource constraints as regards to developing software engineers with up-to-date knowledge who could make strong IT contributions to private and public sector firms. Yet, the data are also consistent with the failure of Japanese corporate, governmental, and educational establishments to recognize and to act upon the importance of software. Furthermore, growing undergraduate CS student and related enrollments faces multiple challenges from non-CS faculty resistance, declining student interest, to minimal industry interest in hiring such graduates. As such, any human resource constraints impacting software innovation can be seen as proximate causes arising from more deep lying structural factors. I address the issue of student interest in IT in the next section.

Software Engineering as an Attractive Career Route

We can begin by examining the results of a CNN/Money Magazine study of the best jobs in America were published in 2010. The survey methodology combined objective (e.g., growth rate of jobs, salaries, educational requirements) and subjective quality of life measures derived from a PayScale survey (e.g., amount of reported job stress, job flexibility) to arrive at their final list.⁴²

These U.S. job rankings were included in a June 2012 IPA report aimed at Japanese software development firms.⁴³ The subject of the report was the comparative standing of the Japanese software industry, especially relative to the U.S. The CNN/Money Magazine survey results show that five of the top ranked twenty jobs in the U.S. are filled by software professionals. Kenji Hiranabe, a leader of the Japanese agile software development movement, successfully pushed for the inclusion of the survey results in the IPA report. He explained to me why:

To Japanese engineers, it is amazing that software developer and other related occupations are so highly recognized in the U.S. It seems that industry, investment, employment, innovation,

education and others factors surrounding IT are mutually reinforcing one another in the U.S. to create positive business outcomes.⁴⁴

Katsutoshi Shintani, Chief Advisor to the Software Engineering Center at the Information-Technology Promotion Agency (IPA), commenting on this same data, stated:

Whenever I mention this data, I stress that Japanese software engineers need to be socially respected and rewarded with high pay as in the case of the U.S. I mean to wake up Japanese management to understand that if their companies are relying on software engineers for growth, then their software engineers need to be rewarded with high pay. Japanese firms also need to display publicly to outsiders the corporate contributions of their software engineers. In so doing, young and talented people will be motivated to work in software development.⁴⁵

These comments suggest that software engineers are not treated all that well in Japanese corporations, nor seen as all that important in society. Consequently there are recruitment problems, especially with regard to getting high school students to consider majoring in IT fields. This would certainly be consistent with the structure of the industry; a majority of Japanese software professionals work in subsidiaries and tiered subcontractors customizing enterprise software. These offer lower pay and more severe working conditions than would be the case than if they were employed by larger corporations doing more creative work. Fewer Japanese software professionals are employed in software product oriented firms, whether start-ups or large corporations as is the case in the U.S. It is in the latter kinds of firms where higher status would more likely be accorded.⁴⁶

Japanese IT engineers hold what are often called the new 3k jobs. This labeling is widely used in the media. The term originated in manufacturing many years ago to describe low status manufacturing jobs characterized by dirty, dangerous and harsh conditions (kitanai, kiken, kitsui). In the world of software, this has been transformed to: lots of overtime work, low wages and stressful conditions (kaerenai, kyuuuyo ga hikui, kitsui). This designation is more likely to apply to those working on customization, maintenance, and upgrading of enterprise software than it is to the higher status embedded software development jobs.

The characterization of IT engineers as low status is on display in a set of three featured articles published in Dec. 2006 in the popular business magazine, Nikkei Business.⁴⁷ Their analysis is supported by survey data from some 2,200 IT engineers.⁴⁸ They found that 90% of respondents reported stress in their work with over 1/3 of those reporting strong stress. The biggest source of stress was quotas and work content. Some 25% reported that they didn't think their work was worth doing. Fifty one percent report their salary was low (though no comparative data was reported). Respondents stated their overtime work averaged 12 hours a week (vs. 10.4 hours per month for all industry employees and 17.5 per month for manufacturing employees) according to the government's Monthly Labour Survey. It may well

be that the perception of low wages is influenced by the extraordinarily amount of overtime employees feel compelled to work relative to other jobs. As required by Japanese labor regulations, overtime work under 60 hours a week, is compensated only at a rate of 1.25 times the regular rate. A major problem reported in the survey was poor motivation of software engineers in their late 30s. Particular sources of concern, in these and other articles and surveys, are found in the firms' lack of clarity about employee career trajectories and prospects. Overall, these findings are consistent with the survey results reported by the Information-Technology Promotion Agency (IPA), the METI IT subsidiary. Nevertheless, the data are not drawn from a random sample so it is, at best suggestive.

Even allowing for exaggeration by these media treatments, they create an image that is hardly enticing to young people contemplating their future careers. This can only contribute to the problems universities and corporations have in recruiting software students and professionals.⁴⁹ Consistent with this interpretation, the survey found that some 73% of respondents believed that IT companies are unpopular among students and young engineers. The large amount of expected overtime work is believed to be a major source of these negative views. Clearly, for many of the best and the brightest, IT is not the career route of choice. In the end, this may be more of a constraint on Japanese software innovation than the education system which is far from best in class but often good enough for many purposes.

None of this is to suggest that the situation in the U.S. is ideal. There is a longstanding and ongoing debate as to whether there are shortages or looming shortages of IT talent.⁵⁰ Those denying shortage claim that those arguing for a shortage misrepresent or misinterpret the data to justify importing low wage docile foreign workers through the H-1B visa program. Moreover, it is argued that the import of foreign IT labor diminishes the prospects for American citizens and reduces incentives for them to enter IT studies. This occurs at the same time that talented foreign students continue to flock to the highest ranked U.S. universities to study CS and related fields. Many of these students have stayed to work for varying periods at U.S. corporations. Whatever the merits of the respective arguments, the U.S. IT industry has had available to them a strong supply of talented IT workers and budding entrepreneurs over at least the last 30 years. The results of a 2012 study of engineering and technology companies founded in the United States between 2006 and 2012 found that 24.3% of these companies had at least one key founder who was foreign born. This was a slight decline from previous decades. There is no counterpart to this phenomenon in Japan.⁵¹

There are relatively few examples of Japanese software firms which have achieved global success and few software heroes celebrated in the popular culture. By contrast, the U.S. job rankings shown earlier are consistent with the presence in the U.S. of a large number of software start-up successes and individual software heroes, entrepreneurs like Bill Gates, Steve

Jobs and Mark Zuckerberg, Jeffrey Bezos, Larry Page and Sergey Brin. The U.S. also has many software companies with global reach like Google, IBM and Oracle. The considerable rewards for their founders' and employees success are well publicized. These cannot help but contribute to the positive view of software related jobs as seen in the broader society.

We can gather more insight into this matter by looking at the research findings of Universum, a firm engaged in employer branding.⁵² They conduct large multi-national surveys of student's preferences for the most attractive potential employers. Business undergraduate students were asked to choose the five employers they would most like to work for. The 2012 U.S. findings show that among the top 20 most attractive employers, software firms accounted for four of the top sixteen employers. Google ranked first, Facebook 12th, Microsoft 14th and Amazon 16th.⁵³ Moreover, Apple, which differentiated itself from competitors to a large extent through its software capabilities and strong integration between software and hardware, ranked second. Were Apple included, five of the top sixteen would be software firms. The comparable Japan rankings shows the first software firm on the top 20 list is Nintendo at number 10 and then Google at number 16, a total of two, with one being an American firm.⁵⁴ In short, we see further evidence here for relatively poor student perception of Japanese IT jobs offering promising career opportunities compared to the U.S.

In the business community, successful CIO's in the U.S. are held up as heroes in the business community. Recognized entities such as Silicon.com and CIO magazine produce annual list of the top 50 CIOs. The more popular Computerworld has its list of Premier 100 CIOs announcing that the winners drive strategy and innovation in top tier IT departments.⁵⁵ In short, software has a far more positive reputation in the U.S. business culture as well and it attracts among the best and the brightest in its universities. There are no well-publicized counterparts in Japan and many Japanese corporations either have no CIO or only a part time one.⁵⁶

With hardware increasingly becoming a commodity, it is software which increasingly adds value to the firm and enables competitive advantage. This being the case, one would expect that software personnel would be increasingly rewarded relative to hardware personnel. There is anecdotal evidence in the U.S. which shows heated competition for software personnel as reports grow that firms increasingly acquire other firms just to gain access to their software talent. Such extreme efforts are uncommon in Japan. What about wages and salaries? If Japanese firms are lagging in the recognition of the importance of software relative to the U.S., then we might expect a larger wage advantage for software personnel relative to hardware engineers in the U.S. compared to Japan.

What is the situation in Japan? First, there is a widespread view among software engineers, and in society at large, that their salaries are low compared to other professionals,

such as hardware engineers. Software specialists, whom I interviewed, tend to discount such claims. The only evidence found in our literature search, provides weak support for a modest advantage for hardware engineers. Again, it may be that the long overtime worked by software engineers relative to other professionals fuels their sense of being disadvantaged overall.

We do, however, have data comparing salaries of engineering professions in Japan and the U.S. which provide indirect evidence of a gap. Nakata and Miyazaki report that Japanese engineering professions' overall salaries (whether using yearly or hourly salaries), are significantly lower than their U.S. counterparts, as measured using either exchange rate or purchasing power parity methods. They also found that salaries of Japanese male software engineers (referred to in Japanese statistical reports as system engineers), relative to their U.S. counterparts, ranged from 48.7 to 57.1 percent depending on whether they used exchange rates or purchasing power parity methods. By contrast, male Japanese general engineers fared notably better in their pay with their pay rising to 56.6 to 62.8 of their U.S. counterparts.⁵⁷ This does suggest that Japanese software engineers are not seen as valued a resource relative to their U.S counterparts as Japanese general engineers are compared to their U.S. counterparts.

Conclusion

This analysis began with a documentation of Japanese IT firm's relative weakness in patentable software innovation. One would do well to keep in mind, however, this does not mean that Japanese IT firms are outliers. It is rather more correct to say that U.S. IT firms, from a global perspective, display exceptional performance in software innovation. Were one to compare Japanese IT firms to those in other advanced economies, Japanese performance most likely would compare more favorably.

My analysis drew upon the findings of Arora and colleagues who analyzed USPTO time series data from 1983 to 2004. Specifically they examined the numbers of software citations in hardware patent applications for both Japanese and U.S. firm originated patents. They concluded that IT innovations are become more software intensive for firms in both countries, that Japanese firms rely less on software capabilities in their hardware inventions than U.S. IT firms and they produce notably fewer software inventions. They also concluded that Japanese IT firms are increasingly lagging behind especially in software intensive sectors.

The researches go on to analyze the cause of Japanese IT firms growing software deficits and conclude that they might well arise from a deficit in software human resources. There are, however, problems with this analysis. First the authors dismiss the literature which suggests a "home country" bias that leads U.S. inventors to have a home court advantage in filing with the USPTO compared to foreign inventors such as the Japanese. It is also quite plausible that Japanese firms select the patents they choose to file in the U.S. from among those with the

most promising export/sales potential to the U.S. and other global markets. This criterion could lead them to patent more software intensive solutions in the U.S. This scenario may have nothing to do with software human resource deficits in Japan being the source of their innovation shortfall.

To further pursue possible sources of Japan's shortfall in software innovations, I examined the role of Japanese and U.S. universities in promoting the new field of information technology and CS. There is evidence that the quality of IT capabilities produced at Japanese universities has greatly lagged and continues to lag that produced at U.S. universities. The slowness in upgrading IT curriculum is notable and it is hard not to label Japan a slow follower in this regard. This can be interpreted to support Arora and colleagues' thesis of the continuing human resource constraints hamper Japanese software innovation. These same facts, however, are also quite compatible with the understanding that the Japanese educational establishment didn't grasp the importance of software to Japan's competitive strength or didn't see it as their responsibility to address this problem and thus failed to act aggressively to mitigate software weaknesses. Indeed, our analysis suggests powerful inertial forces that slowed administrative and curriculum innovation in this area. Moreover, a lack of demand for CS graduates suggests that Japan's relative weakness in software innovation compared to the U.S. does not arise simply from human research constraints or lack of recognition of software's importance but perhaps more from lack of corporate demand for creating innovative software products. This may be a result of large scale manufacturing's commitment to legacy monozukuri hardware manufacturing which demands customized software which transmits and enhances its traditional high quality precision manufactured products. These activities consume a huge proportion of Japan's software human resources.

Of particular concern for Japan's IT future, is the seeming low esteem in which IT work is held in Japan versus the U.S. This is reflected in IT not being seen as an attractive career route relative to the U.S. The reasons for this appear to lie in the limited opportunities for innovative activities and associated rewards for most Japanese software engineers. Instead, most Japanese software professionals are employed as cogs in IT service firms dedicated to customizing enterprise software rather than in established IT using/ IT creating firms or in software startups.

Notes

- ¹ A. Arora, L. Branstetter, and M. Drev, 2012. "Going Soft: How the Rise of Software Based Innovation led to the Decline of Japan's IT Industry and the Resurgence of Silicon Valley," *The Review of Economics and Statistics*, Accepted for publication Manuscript 14150, <http://www.mitpressjournals.org/toc/rest/0/ja>. Accessed June 15, 2012.
- ² Ibid, 9-10.
- ³ Ibid, 16.
- ⁴ Ibid, 30.
- ⁵ Ibid, 8-9.
- ⁶ S. Nagaoka, K. Motohashi and A. Goto, 2010. "Patent Statistics as an Innovation Indicator," (Ed.) Bronwyn H. Hall, Nathan Rosenberg), *Handbook in Economics of Innovation*, Vol. 2, Oxford, Elsevier. B.V., 1100-1101.
- ⁷ D. Lippoldt and P. Stryszowski, op.cit., 23.
- ⁸ W. Steinmueller, 1996. "The U.S. Software Industry: An Analysis and Interpretive History," in David Mowery (ed.), *The International Computer Software Industry*, New York: Oxford University Press: 42.
- ⁹ Y. Nakata and S. Miyazaki, 2011. "The Labor Market for Japanese Scientists and Engineers: Is the Labor Market Externalized? What has Happened at their Workplace?" *Japan Labor Review* (Summer): 100
- ¹⁰ J. Dedrick and K. Kraemer, 1998, *Asia's Computer Challenge*, Oxford University Press, 100..
- ¹¹ D. Costa, 2012. "STEM Labor Shortages," *Economic Policy Institute*, (November 19):4, <http://www.epi.org/publication/pm195-stem-labor-shortages-microsoft-report-distorts/>. Accessed April; 15, 2013. Original source is unpublished data from National Science Foundation's 2003 National Survey of College Graduates provided to the Economic Policy Institute.
- ¹² Y. Nakata and S. Miyazaki, 2011, op.cit.
- ¹³ In 2012, those receiving PhDs in CS taking jobs in industry (61%) or government (4%) together accounted for 65% of all CS PhDs taking jobs from North America in PhD granting departments. This compares to 29%, taking jobs in education from PhD. granting institutions, of which 14% were postdocs. Computing Research Association, 2013. 2012 Taulbee Survey http://www.cra.org/uploads/documents/resources/crndocs/2012_taulbee_survey.pdf
- ¹⁴ F. Rosamund, 2011. "Statistics of the Field," in E. Blum and A. Aho (eds.), *Computer Science: The Hardware, Software and Heart of It*, Springer, New York: 423.
- ¹⁵ <http://www.shanghairanking.com/SubjectCS2012.html>.
- ¹⁶ E-mail communication from Prof. Tetsuo Tamai, Hosei University, Nov. 8, 2012.
- ¹⁷ Ibid.
- ¹⁸ Email communication from Prof. Tetsuo Tamai, Nov. 20, 2012.
- ¹⁹ N. Takahashi, 2001. "Information Processing Society of Japan Accreditation Committee Activities," (Part 1 of 2), (Jouhou shori gakkai ni okeru akuredeiteshiyon iinkai katsudou), *Bit*, (March):8.
- ²⁰ J. Dedrick and K. Kraemer, 1998, op.cit., 48.
- ²¹ Email communication with Michael Cusumano, April 17, 2013. Further confirmation comes from Masato Takeichi, e-mail communication, April 18, 2013.
- ²² Op.cit., N. Takahashi, 1.
- ²³ Ibid., 5.
- ²⁴ M. Takleichi, 2003. Concerning Education of Personnel in Information Field in Japan's Universities (Waga kuni ni okeru jōhō bunya no jinzai ikusei ni tsuite), "2003 Information Studies Symposium" (2003 nen Jōhōgaku Shinpojiumu), Science Council of Japan (Nihon Gakujutsu Kaigi), Information Studies Research Contact Committee, Organizer: 14.

-
- ²⁵ Interview with Prof. Masato Takeichi, National Institution for Academic Degrees and University Evaluation, Tokyo, March 15, 2013.
- ²⁶ F. Kimura, 2013. Symposium on IT and Monozukuri: Complexity of Artifacts and Control Structures Among Mechanical, Electrical and Software Sub-systems," Manufacturing Management Research Center, University of Tokyo, (March 18).
- ²⁷ P. Denning, 2003. "CS: The Discipline," in A. Ralston and. E. Reilly (Eds.), 2nd edition, Encyclopedia of CS and Engineering, New York: Van Nostrand and Reinhold.
- ²⁸ <http://www.t.u-tokyo.ac.jp/etpage/introduction/history.html>.
- ²⁹ Op., cit. Interview with Masato Takeichi.
- ³⁰ Op., cit. F. Kimura, 2013
- ³¹ Op., cit. Interview with Masato Takeichi.
- ³² Computing Research Association, 2013. 2012 Taulbee Survey, http://www.cra.org/uploads/documents/resources/crndocs/2012_taulbee_survey.pdf
- ³³ Computing Research Association, 2013. 2012 Taulbee Survey, http://www.cra.org/uploads/documents/resources/crndocs/2012_taulbee_survey.pdf. Accessed 5/20/13.
- ³⁴ IPA, 2011. Kyouiku Kikan no yori IT Gijutsusha kyouiku joukyou (Situation of IT Engineer's Education at Educational Institutions), Gurobaruka o Sasaeru IT Jinzai Kakuho, --saku ni Kansuru Chousa (Survey on measures which support globalization by securing and educating IT personnel), Tokyo: Information Technology Promotion Agency.
- ³⁵ Email communication Abbas El Gamal, Chair EE, Stanford University, April 9, 2013
- ³⁶ C. Ness, 2013." From geeky to cool: Statistics is Berkeley's fastest-growing major" *Berkeleyan* (April 6). <http://newscenter.berkeley.edu/2013/04/16/from-geeky-to-cool-statistics-is-berkeleys-fastest-growing-major/>.
- ³⁷ Email communication, David Hodges, former Dean UC Berkeley, College of Engineering, April 13, 2013.
- ³⁸ E-mail communication from David Hodges, 4/14/13.
- ³⁹ Op., cit. Interview with David Hodges.
- ⁴⁰ E-mail communication with Masato Takeichi, 4/13/13.
- ⁴¹ Unfortunately the data do not break out the actual number of graduating majors in the two areas. <http://www.t.u-tokyo.ac.jp/etpage/introduction/statistics02.html>.
- ⁴² More specifically, the best jobs were picked according to the following methodology: Researchers created an industry screen using Bureau of Labor Statistics growth forecasts for 7,000 jobs, and identified industries with the biggest increases in jobs requiring bachelor's degrees. They then ranked them by 2008-18 expected growth and pay. They deleted jobs if at least 20% of workers were not in a high-growth industry. They kept fast-growing jobs not specific to any single industry. They then created a job screen and selected jobs that required at least a bachelor's degree, and eliminated those projected to grow less than 10% from 2008 to 2018. They cut jobs with median pay below \$60,000 for experienced workers (except certain high growth job titles). They then eliminated jobs with fewer than 10,000 positions nationwide. The final ranking incorporated data from a PayScale Survey in which 40,000 workers rated their jobs on quality-of-life factors such as stress, job security, and flexibility and short- and long-term employment outlook. They ranked job titles using median pay for seasoned workers, industry and job growth, ease of entry, and estimated employment. <http://money.cnn.com/magazines/moneymag/bestjobs/2010/index.html> . Accessed May 5, 2012.
- ⁴³ Information Technology promotion Agency, op. cit., 22.
- ⁴⁴ Email communication July 23,2012.
- ⁴⁵ Email communication July 22,2012.

⁴⁶ R. Cole, Killing Innovation Softly: Japanese Software Challenges, *Manufacturing Management Research Center (MMRC) Discussion Paper Series*, University of Tokyo. http://merc.e.u-tokyo.ac.jp/mmrc/dp/pdf/MMRC424_2013.pdf

⁴⁷ Nikkei Computer Staff, 2006a. "Future Vision of Measures against 3k: Insecurity is the Problem for Motivation," (3k Taisaku yori Shourai Bijon o: Fuan ga IT Enginia no Yarigai wo Sogu), *Nikkei Computer*, (December): 40-43;

2006b, "Stop Losing Engineers by Supporting their Childcare," (Shussan Ikuji Shien de Jinzai Ryushitsu wo Tomeru), *Nikkei Computer*, (December): 44-47; 2006c. "This is the Kind of Company I'd Like to Work at: Candid Answers from 2200 IT Engineers" (Konna Kaisha de Hatarakitai: IT Enjinia 2200 Nin no Honne,"), *Nikkei Computer*, (December): 48-53.

⁴⁸ The respondent and corporate makeup of this Internet survey are as follows: 65% work for IT vendors, 49% are employed in small and medium sized enterprises, 46% are middle managers, 31% are system engineers, 13% project managers, and 12% application and maintenance engineers.. Average age of sample is 39.3, 1,935 are male and 238 female and. 33% are full time employees.

⁴⁹ Nikkei Computer, 2006c, op.cit.,48-53

⁵⁰ D. Costa, op. cit.

⁵¹ V. Wadhwa, A. Saxenian and E. Siciliano, 2012. *America's New Immigrant Entrepreneurs: Then and Now*, Kauffmann Foundation.

http://www.kauffman.org/uploadedFiles/Then_and_now_americas_new_immigrant_entrepreneurs.pdf Accessed March 2, 2013.

⁵² <http://www.universumglobal.com/> Accessed Nov 23, 2012.

⁵³ Unfortunately, there was no comparative data for engineering students.

<http://www.universumglobal.com/IDEAL-Employer-Rankings/The-National-Editions/American-Student-Survey>

⁵⁴ <http://www.universumglobal.com/IDEAL-Employer-Rankings/The-National-Editions/Japanese-Student-Survey>

⁵⁵ Peppard, op cit., 74.

⁵⁶ Information Technology Promotion Agency (IPA), 2012. "Survey of Factors for Dissemination and Expansion of the Areas of Adoption of Non-Waterfall Model Development," (Hi Wotaforu Kata Kaihatsu no Fukyuu Yoin to Tekiyou Ryouiki no Kakudai ni Kansuru Chosa), IPA: 64.

⁵⁷ Y. Nakata and S. Miyazaki, op.cit., 103.