Software

Ashish Arora, Chris Forman*
Carnegie Mellon University
ashish@andrew.cmu.edu, cforman@andrew.cmu.edu

Jiwoong Yoon
University of Georgia
jiwoongy@uga.edu

February 2007

* Corresponding author. Please direct all correspondence to Chris Forman, at cforman@andrew.cmu.edu. We thank Jeff Macher and David Mowery for their comments and suggestions, and Nicholas Yoder and Kristina Steffenson McElheran for outstanding research assistance.

1. Introduction

The global movement of software services activities (defined to include engineering services and R&D as well as the development of software products) to locations outside of the US is an important and growing phenomenon that has recently attracted widespread attention. Over the period 1995-2002, exports of business services and computer and information services grew at an average annual rate of over 40% in India and at a rate of 20% in Ireland. These changes have received widespread attention within the US and have led to concerns of a "hollowing out" of the American information technology sector, and about potential loss of American technological leadership.

However, despite these changes in the location of production of IT services, there is relatively little evidence of global changes in the location of new software product development. US companies have historically been and continue to be the leading exporters of software products. Moreover, evidence using software patents suggests that inventive activity in software continues to be concentrated in the US. In the short run, the US will continue to enjoy a significant lead over other countries in the stock of highly skilled programmers and software designers that provide it with an advantage in the production of new software products. Moreover, proximity to the largest source of IT demand and potential agglomeration economies arising from proximity to competitors and complementors provide software product companies located in the US with a significant advantage.

2. Dispersion of Inventive Activity in Software

In this chapter we provide evidence on the geographic distribution of inventive activity in software. Economists have long made a distinction between innovation and invention in the study of technological change. Schumpeter (1934) defined innovations as new, creative combinations that upset the equilibrium state of the economy. Mokyr (2002) defines invention as an increment in the set of technological knowledge in a society. Schumpeter pointed out that invention does not imply innovation, and that it is innovation that provides capitalism with its dynamic elements. Because it is more easily measured, in this chapter we will focus on the geographic dispersion of inventive activity. However, we adopt the position of Mokyr (2002), who argues that in the long run invention is a necessary precursor to innovation.

Unlike some of the other industries studied in this volume, one feature of software development is that it frequently occurs both by suppliers of software packages and services as well as by users themselves. As a result, software development occurs throughout all industries in the economy, and so to understand the location of inventive activity in software it is insufficient to examine where one or two industries are located.

To understand this point further, it is helpful to understand further the types of software development activity. The design, installation, implementation, and use of software consist of several phases. Messerschmitt and Szyperski (2002) identify two distinct value chains in software development. First, there is a ***supply value chain*** in which software creators develop software artifacts that provide value for the end-user. This part of the software value chain consists primarily of design and development activities that can be thought of as software "production." In the past this role had been played primarily by independent users, third party programmers, or independent software vendors creating custom software, but over the past 20 years this role has passed increasingly to independent software vendors creating software products.

The output of this value chain contains all of what we would traditionally define as software products, such as word processors, operating systems, enterprise software such as

Enterprise Resource Planning (ERP) and business intelligence software, as well as middleware software that has been productized, such as some transaction processing middleware and enterprise application integration. The total value of production in the software product industry was $61,376.9 million in 1997,[1] and 195.2 thousand persons were employed in this industry in the same year.[2] Firms that operate in this value chain include all of the well-recognized names that are traditionally regarded as "software" firms, including Microsoft, Adobe, Oracle, and the SAS Institute, as well as smaller firms such as Oblix and Primatech.

This value chain also includes the activity of third party firms involved in custom programming and software analysis and design. Such firms create custom software products for their customers, and include firms like CIBER, Inc., Intergraph Corp., and xwave Solutions. The total value created in custom programming and design services was $115,834.6 million in 1997 while total employment was 675.0 thousand in 1997, indicating that both revenue and employment in this sector is greater than that in the packaged software industry.[3] Moreover, custom programming and design services are also growing faster than is the software publishing industry. Though 1997 is the last year for which we have data on revenues by industry, we can compare employment growth across these two industries. Employment in custom programming and design services has grown from 675.0 thousand in 1997 to 1025.3 thousand in 2005, for an average annual growth rate of 5.8%. In contrast, employment in software publishers has grown

---

[1] Source: US Bureau of Economic Analysis Input-Output Tables. This figure includes the total value of products made in NIPA industry 511200 (Software Publishers). 1997 is the latest benchmark year for the Input-Output tables. More recent years do not separate software producers from other information publishers.
[2] Source: Bureau of Labor Statistics (BLS) data on the number of employees in software publishing industry (NAICS 5112), available at http://www.bls.gov/ces/home.htm.
[3] These calculations are based on total sales in custom computer programming services (NAICS 541511) and computer systems design services (NAICS 541512). This latter category may include activities outside of programming, such as IT systems design and integration. A conservative estimate of the value and employment of third party custom programming services uses only NAICS 541511, and yields and estimate of $86,326.8 million and 522.3 thousand, respectively.

from 195.2 thousand in 1997 to 238.7 thousand in 2005, for an average annual growth rate of 2.5%.

Second, there is a software *requirements* value chain in which users add functionality to software to meet their own needs. Users engage in co-inventive activity (Bresnahan and Greenstein 1996) to translate general purpose software into a specific application. Such co-inventive activity may include modifications to packaged software applications or development of new applications. However, in business software it also involves changes to business processes or organization design.

Activity in this value chain includes both programming by professional programmers and software designers employed by IT-using firms, as well as programming activities performed by users themselves. The activity of both groups is difficult to measure, but represents a major share of value created. Scaffidi, Shaw, and Myers (2005) estimate that there were approximately 80 million end user programmers in 2005,[4] compared to 3 million professional programmers. Moreover, occupation data from the US indicates that over two thirds of software professionals do not work for IT firms but rather work for IT-using industries.[5] Neither this software development activity performed by users nor the work performed by software professionals working for IT users is measured in any systematic statistics.

Though systematic evidence is rare, what we do know suggests that economic activity in this value chain is likely to be far larger than that in the supply value chain. According to Gormely et. al. (1998), though the typical cost of implementing an enterprise resource planning (ERP) application suite is $20.5 million, only $4.0 million of this cost is related to hardware and software. The rest is due to the costs of implementing and deploying the software within the

[4] This estimate includes those who create user-developed software that is not sold in markets.
[5] Source: BLS Occupational Employment Statistics.

business. Using data on sales of software products and services in several Western European countries, Steinmueller (2004) estimates that for every €1 spent on software there is an additional €2.36 spent on IT-related business services. However, this estimate is likely a lower bound, since it includes only software services conducted through market transactions, and excludes software development activities within IT-using firms themselves.

The importance of the software requirements value chain has two implications for the measurement of where inventive activity in software takes place. First, a large part of value creation in software takes place outside of firms that are considered to reside in what is considered the software product industry. The value of this activity goes largely unmeasured in traditional governmental statistics, as it often occurs as a labor expense within firms developing or implementing packaged software.

Second, it is very difficult to place a precise definition of what exactly constitutes inventive activity in software. Creation and modification of source code is of course one major component, but so are user modification and business process change. Should these latter activities be included as well?[6] Moreover, how should we treat changes to software code that are embedded in IT hardware? Are these hardware or software inventions? As we will discuss below, given available data, a precise estimate of inventive activity in software is probably not feasible. Instead, we provide a variety of metrics that enable us to estimate broad trends and orders of magnitude in economic and inventive activity in software.

In the section three, we provide evidence of recent trends in globalization of software services. These data provide evidence on globalization of activity in the software requirements value chain and some inventive activity conducted by services firms in the supply value chain,

[6] It is interesting to note that the U.S. patent office has struggled with similar definitional issues, within the context of so-called "business method patents" (Allison and Tiller 2003).

though they will largely miss changes in cross-country software service activities that are undertaken by firms outside of the software services industry. In section 4 we use US software patent data to examine changes in the global dispersion of inventive activity in software product development.

3. Trends in the location of value creation

In this section we investigate broad trends in the location of value creation activities in software. We begin with some statistics describing global variation in the exports and imports in software products and services, and continue with a qualitative description of recent trends in countries that have been known to be active producers in the market for software products and services.

3.1 Statistical trends

Software Products: Figure 1 shows the percentage of total 2002 software *product* exports and imports by selected OECD countries. The figure shows that among OECD countries, the US continues to be the leader by a wide margin in the export of software products, accounting for 21.7% of total software exports. The next closest country is Ireland which accounts for 16.0% of software exports. However, as we will discuss in further detail below, most of Ireland's software exports arise from US multinational companies (MNCs) that utilize Ireland as a base of operations to localize US software products to be shipped to countries in the European Union (EU).[7] Since the bulk of software product exports from Ireland are due to U.S. multinationals in Ireland—Sands (2005) shows that over 92% of Irish software exports are from foreign firms—this suggests that the share of US software exports in global trade flows is probably closer to one third rather than the one fifth that the OECD statistics indicate. Following that, the next largest

[7] Such localization activities include activities such as manual translation or adapting software products to local markets.

exporters are Germany (due in part to software exports from ERP giant SAP) and the United Kingdom. No other country accounts for more than 10% of software exports. Most notably, Japan accounts for only 2.5% of total software exports.

Figure 2 presents total packaged software product sales by region. The story here remains the same: North American represents the largest share of packaged software sales, and this percentage has been increasing over time from 47% in 1990 to 54% in 2001. We explore why other countries have not been more successful in developing software products in further detail below.

Software Services: Figure 3 shows data from the OECD Economic Outlook (2006) that reports the global share of 1995 and 2004 exports in IT services, obtained by summing the categories "computer and information services" and "other business services" from the IMF Balance of Payments data. Though subject to a variety of caveats about measurement and coverage, Figure 3 suggests that the distribution of IT service exports is more evenly distributed across countries than is the distribution of software product exports. Many smaller countries are experiencing rapid growth in their exports of IT services, though some are starting from a very small base.

To explore trends in imports, we use data from the US Bureau of Economic Analysis (BEA) on International Trade in Services. Table 1 provides data on interfirm trade in exports and imports of IT services in 1998 and 2004, calculated by summing the categories "Computer and Information Services" and "Royalties and License Fees." [8] Exports of these services grew from

[8] The columns labeled "Computer and Information Services" provide data on exports and imports of imports of private services among unaffiliated firms. The column "Royalties and License Fees" in the same table includes computer-related services that were delivered to foreign markets through cross-border software licensing agreements. These data do not include intrafirm exports of computer services because BEA does not in general release statistics on many of the countries in Table 1. They also do not include wages of US residents who provide computer services to nonresidents..

$6900 million to $10,862 million from 1998 to 2004, while imports grew from $1992 to $2591 million from 1998 to 2004.

There are three things to notice about this table. First, at present the numbers are small relative to total US trade in services: exports and imports of software services represent 3.3% and 1.0% of total exports and imports of services respectively. Second, the US maintains a positive overall balance in trade and services; moreover, over the period 1998-2004 exports of computer services grew at a faster rate than imports (7.86% versus 4.48% average annual growth rate (AAGR)). Third, although imports of computing services from India grew rapidly from 1994 to 2004, overall US imports from India and the other software underdogs are small relative to other estimates.

Data from other sources suggests that the US data may underestimate imports of software services. An OECD estimate indicates that over 90% of Indian service exports to OECD countries are not accounted for in the data on service imports published by these countries (OECD 2004). Other analyses report similar difficulties in tracking Indian software services exports to the United States. A recent GAO report notes that for 2002, the United States reported $240 million in unaffiliated imports of business, professional, and technical (BPT) services from India, while India reported about $6.5 billion in affiliated and unaffiliated exports in similar services categories (GAO 2005).[9] For 2003, the United States reported $420 million in unaffiliated imports of BPT services from India, while India reported approximately $8.7 billion in affiliated and unaffiliated exports of similar services to the United States. The bulk (40-50%) of the difference, according to the GAO, is because the US does not count the earnings of temporary workers resident in the US in services imports. Other sources include differences in

[9] Affiliated trade occurs between US parent firms and their foreign affiliates and between foreign-owned firms in the US and their foreign parent. Unaffiliated trade occurs between US entities and foreign entities that do not own, nor are owned by, the US entity.

coverage (e.g., embedded software is counted as exports of goods by the US, or IT-enabled financial services are not classified as IT services by the US), and because US data does not indicate affiliated imports by country of origin.

As noted above, services trade data do not capture intrafirm migration of software activity abroad. The US BEA data on US Multinational Companies provide detailed data on the investment and production activities of US companies abroad. Table 2 shows that growth in employment in IT services and computer design industries has been faster for foreign affiliates of US firms than for their domestic operations (AAGR 5.1% v. 3.9%), due to faster growth among foreign affiliates in computer design and related services.

Financing of Software Products and Services: Table 3 includes data on one of the inputs to software product and service firms: financial capital. It includes data on disclosed rounds of venture capital financing by year and by destination country as reported in the Venture Economics VentureXpert database. As is well known, venture financing exhibits significant yearly variation (e.g., Gompers and Lerner 2006) and our data may not capture all venture financing rounds. However, some broad trends are suggested. First, similar to our data on inventive outputs (described in further detail below), the US clearly dominates in inputs of financial capital to emerging software firms. However, based on data from 2002-2005, there is some evidence that rounds of venture financing to the software underdogs declined less from their 2000 peak than did financing to US firms.[10] However, there was an apparent decline in venture financing to these countries in 2005. In short, more years of data are needed to discern whether there is a trend of increasing venture capital financing to the software underdogs.

3.2 Regional Trends in Packaged Software and Software Services

[10] The software underdogs consist of India, Ireland, Israel, Brazil, and China.

In the previous section we showed that the US represents the majority of world sales in packaged software. However, other regions of the world have a large and increasing percentage of software services. In this section we discuss some regional trends that are partially responsible for the geographic variance in economic activity in packaged software and services.

Software Producers in Europe and Japan

In Western Europe, the software industry has long been dominated by custom software development and software services (Malerba and Torrisi 1996; Steinmueller 2004). Table 4 shows sales of software products and IT services in the EU-15 during 2003 - 2005.[11] IT professional services such as consulting, implementation, and operations management are larger than the entire software products market. Malerba and Torrisi (1996) identify several reasons for this focus on software services, including a weak local IT hardware industry, first mover advantages by US software product firms, fragmentation of local demand, and relatively little interaction between European universities and industry. The largest European producer of packaged software is SAP, the producer of enterprise software. SAP is currently the third largest software product company by sales, behind Microsoft and Oracle.

One surprising result in Figures 1 and 2 is that in contrast to many other technology industries, Japanese firms account for a very small share of the total export market for packaged software. This is not a recent result; Japanese firms have not ever been major players in the world market for packaged software, despite their success in video games and in other IT markets. Japan runs a significant negative trade imbalance in software: In 1997, Japan imported US$3.93 billion of software but exported only US$23.33 million (Asahi Shimbun, reported in Anchoroduy 2000).

[11] The EU15 comprised the following 15 countries: Austria, Belgium, Denmark, Finland, France, Germany, Greece, Ireland, Italy, Luxembourg, Netherlands, Portugal, Spain, Sweden, United Kingdom.

A number of reasons have been provided for the relative weakness of Japanese software producers, including challenges created by the Japanese language, weak venture capital markets, weakness in intellectual property protection and weak university computer science education (Anchordoguy 2000; Fransman 1995; Baba et. al. 1996 Cottrell 1996). Cottrell (1996) argues that weakness in Japanese PC software production was due historically to a fragmented standards environment, while Anchordoguy (2000) argues that the above proximate reasons were ultimately caused by Japan's economic system of "catch-up capitalism." [12]

Other Countries that are Large Software Producers

Rapid growth in the size of the Indian software industry has recently attracted much attention in the academic and popular press (e.g., Athreye 2005a; Arora, Arunachalam, Asundi, and Fernandes 2001). Data from NASSCOM show that Indian IT services exports have grown from $22 million in 1984 to $10 billion in 2005, with an additional $3 billion due to R&D services, engineering services and software products. As this makes clear, the Indian software industry has largely been built around software services rather than products. Athreye (2005a) estimates that in 2000 revenue per employee among Indian software firms was approximately $35.1 thousand, up from only $6.2 thousand in 1993.

Some anecdotal evidence suggests that Indian firms are increasingly performing more R&D-intensive activities. Athreye (2005a) notes the growth of a new innovative sector of small niche companies. Moreover, there is evidence of a deepening of R&D skills and the emergence of informal networks among local firms in India. This is also some evidence of success in certain niche technologies such as wireless and embedded systems (Parthasarathy and Aoyama 2006;

[12] In particular, she argues that some of the key elements of the Japanese economic system—including state targeting policies, its keiretsu industrial groups, bank-centered financial system, and weak intellectual property system—have been benefited its development of successful industries in steel, semiconductors, and IT hardware but have hindered the development of its IT software industry.

Ilavarasan 2006); software for mobile phones represents a substantial category. Some Indian firms have also had success in developing software products for the developing countries market: one example is CITIL (now i-flex), a Citibank subsidiary that initially produced software products for developing country markets before eventually moving on to head to head competition with the established incumbent producers in developed countries (Arora 2006; Athreye 2005b). There is also some data on substantial and growing R&D activities in countries such as India; Arora (2006) reports that total revenues for engineering services and R&D by Indian producers in 2006 was estimated to be US$4.8 billion, a 23.1% increase over the prior year. In the next section we attempt to shed some additional light on this issue by examining US patent data.

The Irish software industry consists of two very separate sub-industries, each with very different characteristics. First, there is an overseas sector that is dominated by multinational firms (MNCs). These firms primarily are engaged in software logistics (such as media replication and printing and packaging production and distribution), localization (such as translating and adapting software to suit European markets), and development (O'Riain 1997). Second, there is an indigenous sector that is populated by smaller firms that is engaged in software development and product development activities.

The number of MNCs in Ireland grew rapidly throughout the 1990s, from 74 foreign firms in 1991 to 140 foreign firms in 2000. As Arora, Gambardella, and Torrisi (2004) note, this rapid growth was due to a number of factors, including the liberalization of economic policies that began in 1991; a large and well-educated English-speaking workforce; an advantageous site for localization activities; as well potential agglomeration economies that were ignited after the Irish software-producing industry reached sufficient scale. MNC subsidiaries are engaged primarily in “low-value-added, low-skill activities such as porting of legacy products on new platforms, disc duplication, assembling/packaging, and localization” (Arora, Gambardella, and Torrisi 2004). Revenues and exports in the Irish software industry have always been dominated by these MNCs. Sands (2005) notes that total industry revenues grew from $2.66 billion in 1991 to over $18 billion in 2002, with MNCs continuously accounting for over 90 percent of the total. In contrast, the indigenous sector is more product-based: it accounts for just under half of employment, however it accounts for only 9 percent of revenues. Indigenous companies are usually young and small, and often produce primarily for niche or vertical (i.e., industry-specific) markets (Sands, 2005).

The software industry in Israel looks considerably different from that in either Ireland or India. Compared to locally-owned Indian or Irish firms, Israeli firms are more product-based and are more R&D intensive. Bresnitz (2005) notes that revenue per employee for Israeli software firms was US$255,172 in 2000. By his calculations, the similar statistic in 2000 for US software publishers was US$231,621 and for locally-owned Irish software producers was US$90,000.

Breznitz (2005) examines the reasons for Israel’s product-based industry. He provides several reasons, including: tight links between the R&D activities of Israeli universities and high-tech industries in the country; the presence of a highly successful indigenous hardware industry; the presence of local market demand for new products; the presence of American MNCs locating R&D facilities in Israel; and the ability of the Israeli IT industry to raise capital in US financial markets.

4. Empirical Evidence on the Location of Inventive Activity

In this section we examine the global geographic distribution of inventive activity in software. The data presented in the prior section pointed to expanding markets for software services abroad. Those data also show that the market for packaged software continues to be highly concentrated in the US, and little evidence indicates that this trend is reversing. However, authors such as Athreye (2005a) report increasing inventive activity in Indian firms, and other authors have reported similar trends in Ireland (Sands 2005) and China (Tschang and Xue 2005), as well as well-established software product industries in Israel (Breznitz 2005) and Brazil (Botelho et. al. 2005). Software product sales are a lagging indicator of inventive activity in software: could inventive activity in software be picking up in other areas of the world but not yet reflected in product sales? If so, how significant are these developments in terms of number of inventions and their importance? To answer these questions, one needs a measure of R&D and inventive activity that is comparable across countries.

Patent data have long been used as one measure of inventive activity. Patents have also been found to be correlated, although weakly, with R&D spending, so they provide a weak measure of raw inputs into innovation (Griliches 1990). There are, of course, significant limitations to the use of software patents as a measure of inventive activity. As Jaffe and Trajtenberg (2002) note, not all inventions meet the US Patent and Trade Office (USPTO) criteria for patentability,[13] and inventors must make an explicit decision to patent an invention, as opposed to relying on some other method of intellectual property protection. Both of these issues are particularly acute in the patenting of software. Historically, inventions in software were not patentable[14] and for a time copyright was the predominant form of formal intellectual property

[13] Note that not all inventions also meet the criteria for patentability for the European Patent Office (EPO) and Japanese Patent Office (JPO).
[14] The following provides necessarily a brief overview of the history of intellectual property protection in software. For a more detailed overview, see Graham and Mowery (2003) and Hall and MacGarvie (2006).

protection in software. However, a series of court decisions widened the scope of software patents. Eventually, this culminated in the Commissioner of Patents issuing guidelines for the patenting of software that allowed inventors to patent any software embodied in physical media (Hall and MacGarvie 2006). In contrast, over the same period a series of cases, including several copyright infringement cases brought by Lotus Development, weakened the intellectual property protection offered by copyrights. Graham and Mowery (2003) show that over this period the number of granted software patents has increased dramatically while the propensity of firms to copyright has declined.[15] Recent research has shown that the stock of patents is correlated with firm success in the software industry (Merges 2006), suggesting that patents may be a potentially useful metric of the inventive output of firms.

A second issue in using software patents to measure inventive activity in software is identifying exactly which patents are software patents.[16] Software patents are not assigned to a particular class or subclass in either the USPTO or International Patent Classification (IPC) schemes. Moreover, there is no unique field in patents identifying them as software patents. Graham and Mowery (2003) were the first to systematically identify software patents for research purposes. They identified the International Patent Classification classes used by the six largest producers of personal computer software over the period 1984 to 1995. This search resulted in a list of 11 IPC classes, which account for over half (57%) of the over 600 patents assigned to the 100 largest packaged software firms in 1995 (as identified in the trade news publication, *Softletter*).

---

[15] The set of patentable inventions is narrower in Europe than in the US. To be patentable, then European Patent Convention requires that inventions address a particular technical problem and suggest a technical means to solve this problem (Thoma and Torrisi 2006). The implication of this requirement is that "inventions having a technical character that are or may be implemented by computer programs may well be patentable" (EPO 2005).

[16] This section provides an overview of the issues in identifying software patents. For a more complete discussion, see Layne-Farrar (2005) and Hall and MacGarvie (2006).

The Graham-Mowery approach of using the patent classification system to identify software patents has been used and revised by others. Graham and Mowery (2005) identify software patents using USPTO classifications. Hall and MacGarvie (2006) identify software patents by finding the USPTO class-subclass combinations in which fifteen large software firms patent. To identify their final sample, they intersect the resulting set of patents with another keyword definition used by Bessen and Hunt (2004).

Bessen and Hunt (2004) identify software patents through the use of a Boolean query that searches for keywords in the text of patents. They arrive at a patent sample that is broader than that used by other researchers (Layne-Farrar 2005). Other researchers have identified a smaller sample of patents by reading them manually. Allison and Tiller (2003) identify Internet business method patents and Allison et. al. (2005) identify university software patents.[17]

For this paper, we use a version of the Graham-Mowery approach based on the IPC classification system. We began by identifying the top 10 firms by revenue volume in 1995 according to the Corptech Directory of Technology Companies.[18] We then examined which IPC classes they patented in. Because we found that the Graham-Mowery set of IPC classes covered only 46% of the patents of these top 10 firms, we added two additional IPC categories. Our complete list of patent classes covered over 80% of the patents of these top 10 firms. Table 5 provides a list of the included IPC classes and subclasses and their descriptions.

By using a broader set of IPC classes than Graham-Mowery, we are more likely to include patents which may be assigned to the above classes but which are not software patents. As we will see, software patenting outside of the US is relatively rare, so we utilize a conservative definition that includes as many such patents as possible in hopes of achieving an

[17] Thoma and Torrisi (2005) compare several of these methods in a study of European software patents.
[18] These are Adobe, Autodesk, Cadence, Macromedia Inc, Microsoft, Novell, Oracle, SAP, Sybase, Symantec Corp

"upper bound" on the stock of software patents invented outside of the US. However, we recognize that if the rate of patenting in related technologies outside of software is higher than that inside it and if the share of inventive activity in these other technologies is higher in the US than in abroad, then our measure may artificially inflate the gap in software patenting between the US and other nations. To address this possibility, we compare our results using several software patenting definitions: including Graham-Mowery (2003) and Graham-Mowery (2005).

As an illustration, we computed the percentage of patents produced by inventors who reside in the US (regardless of assignee location) under different definitions and then compared them. Figure 4 presents these results. All three definitions show similar percentages for US patents. Moreover, the three definitions have similar trends: increasing throughout the 1990s before reaching a peak around 2000 before declining slightly. Given the similarity in results across these different definitions, we will continue to focus on our original definition described above.

4.1 Results of Patent Data Analysis

Figure 5 shows the number of US patents invented in the US, Japan, other G-7, and all other nations (based on inventor address) by year of patent grant. The steep increase in the number of patents granted post-1995 is consistent with prior work that has shown an increase in the propensity to patent software after increases in the scope of intellectual property rights afforded by software patents (Graham and Mowery 2003; Hall and MacGarvie 2006). In 2004, 4695 software patents were issued to inventors in the US, a larger number of patents than all other areas of the world combined (2811). The average annual growth in software patenting between 1988 and 2004 was also greater in the US than in all other G-7 nations: patenting by US inventors grew at an average annual rate of 19.5%, compared to 16.1% in Japan and 18.0% in other G-7 nations.

These figures may reflect a “home country bias”: US firms may be more likely to patent in the US market than foreign firms. Thus, in our data on patenting by location of inventor, the high percentage of US patents may reflect (1) higher rates of US patenting by US firms (compared to firms in other countries) and (2) a higher propensity for US firms to invent in the US. More broadly, there may be some concern that there are potential differences between the site of inventive activity in US-assigned US patents that have EPO or JPO equivalents and the site of inventive activity in US-assigned US patents that do not have such equivalents. We address this potential concern in two ways. First, we look at the location of inventive activity for patents assigned to firms from outside of the US. Second, we compare our results to recent work that has examined software patenting behavior in European patents.

We examined the percentage of patents assigned to the home country by country of assignee firm, based upon year in which the patent was granted. Figure 6 shows that Japan-assigned US software patents are predominantly invented in Japan, although this share appears to decline during 2000-2004. Similarly, the location of invention in Israeli- and G-7 assigned patents (excluding the US and Japan) is predominately sited in those countries and regions. To be clear, comparing the propensity of US software patents assigned to US firms to be invented in the US with the propensity of US software patents assigned to firms from other countries to be invented in that (home) country is not an “apples to apples” comparison. However, given this important caveat, this figure does not suggest that patents assigned to US firms are significantly more likely to be invented in the home country (US) than are the patents from other countries. In fact, for several years, the proportion of US patents are assigned to Japanese and Israeli firms are more likely to be invented in the home country than US patents assigned to US firms. In recent years, however, this “home” percentage has been higher for patents assigned to US firms than for others, though this is largely attributable to a decline in the home invented share for patents assigned to firms from other countries.

Thoma and Torrisi (2006) examine the rate of software patenting in European patents. Figure 7 shows the number of patents granted by country of patent assignee and year of patent application. There are some differences in the way Thoma and Torrisi define software patents and other differences in their sample construction: in particular, Thoma and Torrisi examine the distribution of patenting activity by site of assignee rather than inventor.[19] However, the broad trends are very similar to those in Figure 5: US firms are responsible for the majority of software patenting activity, followed by Japanese firms, and then all others. Moreover, Thoma and Torrisi (2006) note that of the European software patents in their database, 80.3% have also been granted by the USPTO and 73.8% have also been granted by JPTO. If the majority of European software patents assigned to US firms are also invented in the US (not an unreasonable assumption given evidence presented in the earlier paragraph that the majority of US software patents assigned to US firms are invented in the US), then the graph suggests that even using European software patent data a large share of the inventive activity in software takes place in the US. Further, we note that while the levels of software patenting expressed in Figures 4 and 5 may be influenced by home country bias, so long as this bias does not change systematically over time, the time-trends shown in these figures will not be as influenced by such bias.

[19] In particular, Thoma and Torrisi use a variant of the Hall and MacGarvie (2006) method of constructing a software patent sample based on patent classes and Bessen and Hunt's (2004) key word method. Moreover, this graph shows patenting by assignee country rather than inventing country, however according to our data 93.4% of patents assigned to US firms were also invented in the US. Last, this figure shows patenting by year of application rather than year of granting, however the broad trend of greater patenting among US assignees is robust to this difference.

Patenting Activity by Region

Figure 8 shows the number of US patents invented in the underdog countries based upon inventor location. Israel is the only one among them to have a significant number of US patents. Israeli patenting activity increased from 3 in 1998 to a high of 90 in 2003. No other country has had more than 20 patents in any one year, though the number of patents invented in India has risen slightly in recent years, from an average of 0.5 throughout the 1990s to 16 in 2004.

Figure 9 shows the number of patents invented in the East Asian Tigers based upon inventor location.[20] The number of patents invented in these countries is significantly higher than that of the underdogs. However, evidence suggests that many of these patents may be electronics-related.[21] Patenting among these countries is dominated by inventors from indigenous electronics companies in Korea and Taiwan: in 2004, 264 of the 280 patents granted were from this set of assignees.

Assignee Location for Patents Invented Abroad

As noted earlier, multinational firms have played a major role in the development of software industries in other countries such as India and Ireland, and may be driving the patenting activity by overseas inventors. To investigate this question further, we examined the location of US software patent assignee for US software patents invented in different countries. The overwhelming majority of patents invented in the US were also assigned to US firms. This fraction ranges from 93% to 97% over the period 1988-2004. No other region ever exceeds 6% in these data.

[20] For the purposes of this paper, the Asian Tigers consist of Korea, Taiwan, Singapore, and Hong Kong. These are a separate and distinct set from the software underdogs.

[21] The top patenting firms in these countries include: Daewoo Electronics Co. Ltd. (33); Electronics and Telecommunications Research (60); Hyundai Electronics Industries Co. Ltd. (57); Industrial Technology Research Institute (55); Inventec Corporation (25); LG Electronics (102); Samsung Electronics (463). All of these companies are heavily involved in electronics research..

Figure 10 shows the distribution of assignee country for patents invented in the underdog countries. Here, the fraction of patents assigned to US firms has generally been increasing over time, ranging from 20% in 1990 to a high of 65.7% in 2002. Excluding Israel from the software underdogs (which has a robust software product industry), the top assignees in the software underdogs are 3Com (12), IBM (25), and Texas Instruments (12): no other company has more than five patents. The percentage of patents invented in underdog nations that are assigned to underdog firms has similarly been declining over time, from 80% in 1990 to 32.7% in 2004.

The increasing share of patents invented abroad in one of the software underdogs but assigned to US firms suggests that there may be some shift in the location in inventive activity for US firms to offshore locations. There is some evidence of a shift to more offshore invention for patents assigned to US firms. However, the shift is small and offshore software invention in underdog countries by US firms accounts for a very small share of the total patents assigned to US firms. We also examine the trends in the site of inventive activity for US software patents assigned to US firms: note that these trends, because they only examine the site of inventive activity for patents assigned to US firms, are not subject to concerns of home country bias. The percentage of US assigned patents invented in the US fell from 93.5% in 1996 to 92.1% in 2005.[22] This decrease in the share of US patents is due in large part to the increase in offshore activity in the underdogs: the percent of US assigned patents invented in the underdogs rose from 1.1% in 1996 to 1.8% in 2005.

We next examined whether there were any systematic differences in the industrial classification of the patent assignees by region where the patent was invented. To do this, using assignee (company) name we merged our US software patent data with the Corptech Database of

[22] The share of US assigned patents invented in the US was 96.2% in 1988 and 93.0% in 1989, though the number of software patents in these years was much lower than in 1996 (260 in 1988 and 387 in 1989 compared to 1519 in 1996) which, as described above, was one of the first years in which software patenting began to grow rapidly.

High Technology Companies. Table 6 shows the distribution of assignee industry for patents by inventor region. Due to the way the Corptech data are collected, the industries of many Asian and G-7 countries in the Corptech database are classified as holding companies, so we focus our analyses on patents invented in the US and in the underdog countries. One fact that is immediately apparent across all rows of the table is that patents are assigned to companies that belong to a variety of industries. Outside of the “Holding Company” category, most patents are from the “Other” industry. Moreover, most patents are not assigned to firms in the "Software Publishers" industry (SIC 7372), the SIC industry for packaged software producers. Second, the distribution of industries in the US and underdog countries are broadly similar, with US firms slightly more likely to be in Industrial Machinery and Equipment and the underdogs more likely to be in Electronics.

In Table 7 we provide some descriptive statistics on top patenting firms in major software-producing countries. To construct this table, we identified the five firms with the largest number of US patents in each of 9 countries: China, Germany, the United Kingdom, Ireland, Israel, India, Japan, South Korea, and the United States. Two major facts emerge. First, as noted above, the top patenting firms in software are usually not packaged software producers. Second, the top patenting firms in the underdog countries are usually large US producers of electronics—and to a much lesser extent European and Japanese producers—such as IBM, Intel, Texas Instruments, and Sun Microsystems. One exception is China, where one Taiwanese and one Chinese firm is included among the leading producers. However, as noted above, the number of US software patents produced in China is very small.

5. US Market Advantages for Innovative Activity

The data in the prior two sections show two very different stories going on in the globalization of software activity. On one hand, as has been well-documented, there has been increasing growth in the production of IT service activity outside the US. This trend has been going on for some time now and shows no signs of abating. Second, there is evidence that inventive activity in software development (at least as measured by patents) is highly concentrated in the US and heavily controlled by US firms. Though there is some evidence that inventive activity is picking up outside the US, at current rates of growth it will not catch up with the US software industry any time soon. Moreover, though there is some evidence that some inventive activity by US firms has shifted abroad, at present the shift is small and this remains a small share of US firms' overall inventive activity.

However, these trend rates of growth can change, so it is useful to examine the conditions that are widely thought to be conducive to innovation and inventive activity in new technologies. A long literature has examined some of the factors influencing the variance in innovative activities across countries.[23] These include R&D investments and human capital (e.g., Romer 1990), supportive public policies (e.g., Nelson and Rosenberg 1994; Mowery and Rosenberg 1998), and more localized factors supporting the growth of clusters, including spillovers and user-producer interactions (Porter 1990). In general, the US has advantages over other rich and poor countries in all of these dimensions. We focus our attention on one area that we believe has received insufficient attention: the importance of geographic proximity to lead user innovation.

A key factor in the development and growth of a local software industry is the relationship with users. The transition of new inventions to usable economic products is a difficult process. Solving the problems that remain after initial conceptualization requires sustained innovative activity. User innovation and input is often an important part of this process

[23] For a recent overview and review of this literature, see Furman et. al. (2002).

(Rosenberg 1963), and the willingness and ability of individuals to acquire and use new products and technologies is often as important as the developments of such products and technology themselves (Rosenberg 1983).

Such user activity is particularly important in software. Business software in particular is often bundled with a set of business rules and assumptions about business processes that must be integrated with the existing business organization, its activities and its processes. Recent research indicates that proximity between software developers and users is particularly important for this activity to occur.

The software industry has a long history of user innovation and interactions with users leading to path-breaking new products. For example, IBM's collaboration with American Airlines on the SABRE airline reservation system in the 1950's and 1960's was an important early use of information technology to "real-time" applications that would later be used in airline reservations, bank automation, and retail systems (Campbell-Kelly 2003; Copeland and McKenney 1988). The genesis of this project was a serendipitous event: the chance meeting on a flight of R. Blair Smith of IBM's Santa Monica sales office with C.R. Smith, American Airlines' president. The eventual outcome of this project was the SABRE system. Both IBM and American Airlines made extensive investments and contributions to the project: "We tapped almost all types of sources of programming manpower. The control (executive) program was written by IBM in accordance with our contract with them. We used some contract programmers from service organizations; we used our own experience data processing people; we tested, trained, and developed programmers from within American Airlines, and hired experienced programmers on the open market."[24] Similarly, the early development of ERP software by SAP

[24] R.W. Parker, "The SABRE System," *Datamation*, September 1965: 49-52, as quoted in Campbell-Kelly (2003).

occurred through a series of incremental improvements when developing real-time software for clients (Campbell-Kelly 2003).

One major challenge to offshoring software product development work will result from the difficulty of coordinating software development activity across a globally distributed team. As is well known, partitioning complicated software development projects across multiple team members is difficult, and often substantially increases the costs of software development (Brooks 1995).

These problems may become still greater when attempting to manage such projects at a distance. Globally distributed team members do not have access to the rich communication channels that co-located developers have. Moreover, differences in language and culture may make it much more difficult to establish common ground among team members and ensure that miscommunications do not occur (Armstrong and Cole 2002; Olson and Olson 2000). These projects face other challenges as well, including an inability to engage in informal communication as well as the difficulty of managing team members who may believe that such projects are a prelude to job cuts.

A number of techniques have been proposed for lowering the costs of distributed software development. Going back as far as March and Simon (1958), one common technique in distributed development is to reduce the interdependencies among software components. The increasing modularization of software code and the use of object-oriented software development techniques has likely reduced some of the costs of distributed development over time. However, schedules and feedback mechanisms are necessary when interdependencies are unavoidable (March and Simon 1958). The recent successes of large-scale open source projects such as Linux and Apache have led some to consider whether open source project management methodologies could be utilized in traditional corporate software development. Globally distributed teams rely heavily on coordination tools such as e-mail, phone, and more recently instant messaging as well as configuration management tools. However, several authors have shown that initial meetings are often necessary to both detail project requirements and for project members to become familiar with one another (e.g., Herbsleb et al 2005). In general, the literature has demonstrated that despite the continued development of tools and techniques to manage distributed projects, globally distributed work is difficult and can involve significant coordination costs.

Despite the considerable work that has been done in examining the challenges of software project management in a distributed environment, there has been heretofore relatively little systematic widespread empirical evidence on how distance from software suppliers impacts firm decisions to offshore software development.

Arora and Forman (2006) attempt to gather such systematic evidence by examining which IT services can be effectively performed from a distance or, to put it another way, which IT services are tradable. One way of examining the tradability of IT services is to examine the extent to which they are clustered near local demand. If markets for IT services are local, then we should expect the entry decisions of IT services firms will depend in part upon the size of the local market. If markets are not local, then the composition of local demand should matter little: rather, suppliers should locate in low-cost regions. By providing evidence of the geographic reach of markets, this analysis also provides evidence on the tradability of services: markets for services that are not tradable will be local, while those for services that are tradable need not be local.

Arora and Forman examine the clustering of local market supply for two types of IT services: *programming and design* and *hosting*. "Programming and design" refers to programming tasks or planning and designing information systems that involve the integration of computer hardware, software, and communication technologies. These projects require communication of detailed user requirements to the outsourcing firm in order to succeed. Hosting involves management and operation of computer and data processing services for the client.[25] After an initial set-up period, the requirements of such hosting services will be relatively static and will require relatively little coordination between client and service provider. Thus, ex ante we would expect that hosting activities may more easily be conducted at a distance than other activities. Using data from US Census County Business Patterns, Arora and Forman (2006) find that the elasticity of local supply to local demand characteristics is higher for programming and design (0.806) than for hosting (0.1899). That is, a 10% increase in local market demand will translate into 8.1% increase in the supply of programming and design firms, but only 1.9% increase in the supply of hosting firms.

Arora and Forman also examine whether firm decisions to outsource programming, design, and hosting services depends upon local market supply. Table 8 shows how 2002 outsourcing varies by the size of geographic area in the US. Average outsourcing of programming and design is clearly increasing in the size of a location, though the pattern for hosting is less clear. Outsourcing of programming and design increases from an average level of 24.2% in small MSAs and rural areas to 26.1% in medium and large MSAs, and these levels are significantly different from one another at the 1% level. In contrast, outsourcing of hosting declines slightly from an average level of 15.61% in rural areas and small MSAs to 15.60% in medium and large MSAs; these levels are not statistically different from one other. Since the supply of outsourcing establishments is increasing in location size, these results suggest that the

[25] While hosting activities do not fit most definitions of "innovation" or "invention" in software *per se*, they do provide a useful benchmark to compare tradability of services that require complex communication and coordination between supplier and customer and those that do not.

decision to outsource programming and design is increasing in the local supply of outsourcing firms. Controlling for industry differences, establishment size, and other factors yields the same conclusion.

This evidence, combined with that on the costs of distributed software development described above, suggests that proximity to users is an important determinant of inventive activity in software. The contrast with other products and industries in this volume is informative. For other products such as wireless devices or PDAs lead users have significant concentration in locations outside of the US such as East Asia. However, the lead users of software are predominantly large organizations, and the leading large organizations in use of software and IT remain in the US. This is especially true for the large market segment of business applications software, for which software products and services are frequently embedded in business process. User requirements in this setting often involve the transfer of tacit knowledge, and so proximity to lead users is particularly salient. Thus, as long as the US remains the major market for software products, and the locus of the vast majority of lead users, the US is unlikely to lose its technical leadership.

6. Some Recent Trends and Projections for the Future

Trends in computer science education

Continued success in any innovative industry like software requires a talented and highly educated workforce. There is widely reported concern about a perceived shortage of domestic born scientists and engineers in the US (e.g., Ricadela 2005). Figure 11 shows data from the National Center for Education Statistics (NCES) on the number of undergraduate and master's degrees in computer science earned in the US over the period 1983 to 2002.[26] Both

[26] Data from the NCES and other official government statistics in this subsection is from the National Science Foundation publication *Science and Engineering Indicators*.

undergraduate and master's degrees rose sharply from a combined figure of 35.2 thousand in 1996 to 65.7 thousand in 2002. This increase was influenced by the boom in the information technology sector in the late 1990s.

More recent indicators of undergraduate and master's-level enrollments in computer science are currently unavailable using official US statistics. Figure 12 presents data from an annual survey of incoming freshman. Mirroring the NCES statistics, these data show intention to major in computer science rising throughout the late 1999's and remaining high until 2001. However, intentions to major in CS drop sharply thereafter. The Computing Research Association's Taulbee Survey shows similar findings. These data survey Ph.D.-granting institutions in the US. Analysis in Aspray et al (2006) argue that data from the Taulbee survey closely matches trends in the NCES data, and so these data are a good leading indicator of the national educational statistics. Figure 13 shows a sharp decline in newly declared computer science majors after 2000.

Somewhat more recent official data is available for doctoral degrees conferred by US universities. Figure 14 shows the number of doctoral degrees earned in computer science and mathematics during 1983 - 2003. In contrast to bachelor's or master's degrees, the number of doctoral degrees granted has generally been on the decline in the US over the last decade. The figure shows that the number of computer science PhDs peaked in 1995 at about 1,000, and then has fallen over time. In 2003 the number of such degrees advanced slightly from 810 to 870. However, due to the very long lag between entry and graduation in doctoral programs, this increase likely reflects enrollment decisions in the middle to late 1990s, when demand for computer scientists was particularly strong.

While the number of students entering computer science programs appears to have fallen recently, there is evidence that such enrollments have been picking up in other countries. Figure 14 also shows the number of doctoral degrees granted in mathematics and computer science in selected countries other than the US. The number of doctoral degrees in computer science and mathematics has recently been increasing in Asian countries such as China, Korea, and Taiwan.[27]

Unfortunately, similar statistics are not easily available for the production of bachelor's and master's degrees. Gereffi and Wadhwa (2005) provide evidence on the number of bachelor's and sub-baccalaureate engineering, computer science, and IT degrees for the US, India, and China in 2004. Figure 15 shows that the number of degrees awarded in engineering by India and the US are roughly similar. While the numbers of engineering graduates in China are much larger than that of either the US or India, Gereffi and Wadhwa (2005) note that educational statistics on engineers from China include degrees from two- or three-year programs that include students graduating from technical training programs that may be qualitatively different from baccalaureate programs in the US. When normalized by population, the US continues to lead in the production of bachelor's degrees in engineering, producing 468.3 bachelor's degrees per million compared to 103.7 in India and 271.1 in China.

However, recent work by Arora and Bagde (2006) show that the number of engineering baccalaureate degrees awarded in India is growing much faster than in the US. Table 9 shows that although the number of engineering baccalaureate degrees awarded in 2003 is roughly the same as that reported by Gerrifi and Wadhwa, this number has grown steeply over time. From

[27] These statistics, presented in *Science and Engineering Indicators* and collected from a variety of places, are unfortunately available only with some lag, and may not be strictly comparable. Moreover, they do not provide educational statistics on computer science graduates for India.

about 42,000 in 1992, this has more than tripled to more than 128,000 in 2003.[28] Moreover, since the number of baccalaureates produced reflects the capacity added with a four year lag, it is important to note that sanctioned engineering baccalaureate capacity in India now exceeds 440,000, although a substantial portion is of dubious quality.

Figure 16 shows that the number of foreign students enrolled in graduate computer science programs in the US declined in 2003 for the first time since 1995, reflecting visa restrictions imposed after September 11, 2001, the growth in degree-granting programs in other countries, as well as declines in the demand for engineers and computer scientists that took place in the early years of the most recent decade (NSF 2006).

Overall, the data show that the US continues to maintain a lead in the production of computer science graduates at all levels. However, recent data suggest that enrollments in computer science may be declining in the US and picking up in other nations. As we will show in the next section, however, these changes in domestic supply are likely not due to long run declines in the demand for computer science graduates within the US.

Labor Market Trends

There is some evidence that growth in the number of computer science degrees awarded over the past 25 years has not been fast enough to keep pace with demand for workers with computer science training. Figure 17 shows that the annual growth rate in the production of all mathematics and computer science degrees averaged 4.2% during the 1980 - 2000 period, significantly less than the average annual growth of 9.3% in occupations directly associated with these fields.[29] These data are now several years old and do not account for students receiving

---

[28] These numbers are based on data reported by 14 states, which include all the major states except Bihar, and probably represent 80-90% of the engineering baccalaureates produced in India.
[29] Occupational data from these figures was compiled by the National Science Foundation, Division of Science Resources Statistics, from US Census data.

degrees from outside of computer science but moving into computer science professions. However, despite these qualifications, they do suggest that the US may have relied in part on workers from abroad to make up for the shortfall of native workers with computer and math skills.

Recent data suggests that the inflation-adjusted median salaries for master's graduates in mathematics and computer science rose 54.8 percent between 1993 and 2003, higher than any other broad class of science and engineering graduates and higher than the average across all non-science and engineering graduates.[30] Growth in salaries was similarly competitive for graduates with bachelor's degrees (28.0% AAGR, second only to engineering graduates among science and engineering graduates) and those with doctoral degrees (18.6% AAGR, second only to graduates in engineering and physical sciences among science and engineering graduates). Further, 2003 median salaries for computer science master's graduates are higher than any other broad category of science and engineering graduates ($80,000), while levels for bachelor's ($50,000) and doctoral ($67,000) degree graduates remain similarly competitive. Thus, even when one uses data that includes the recent technology downturn, salaries of occupations requiring skills in mathematics and computer science have remained quite competitive when compared to other occupations in science and engineering and compared to the national average.

As noted above, there has been a significant shortfall in the rate of granting of computer science degrees relative to the rate of employment growth, and this excess demand for workers with computer science and engineering skills has been partially offset by immigration of skilled workers from abroad. In fiscal year 2001 there were 191,397 H1-B visa admissions to the US from computer-related occupations, 57.8% of total such admissions and the largest of any such

[30] The source for these data is the National Science Foundation, Division of Science Resource Statistics, National Survey of College Graduates.

category.[31] Kapur and McHale (2005a) list the top companies that petitioned for H-1B visas in October 1999 through February 2000, a list which includes some of the leading IT hardware and software firms: Motorola (618 petitions), Oracle (455 petitions), Cisco (398 petitions), Mastech (398), Intel (367), Microsoft (362), Rapidigm (357), Syntel (337), Wipro (327), and Tata Consulting (320).

Changes in immigration represent one mechanism that has the potential to impact the US software industry in the relatively short run, and recent changes in the environment outside the US can potentially affect immigration flows. The rapid growth in the software industries of countries like India and Ireland has increased the attractiveness of those countries to highly skilled indigenous workers. This has been particularly evident in Ireland, where rapid growth has encouraged an increasing number of highly skilled workers to remain in Ireland or return to Ireland from the US. Kapur and McHale (2005a) report that emigration of male Irish graduates fell from about 25% in 1987 to under 15% in 1997, with similar trends for female graduates. Of the 644,444 Irish who had spent one year outside of Ireland in a 2002 Census, 42 percent reported taking up residence in Ireland between 1996 and 2002, suggesting that a large fraction are recently returning Irish (Kapur and McHale 2005a).[32]

With the continuing growth of the software industries in India and Ireland, it is likely that these historically important sources of highly skilled software professionals will retain a growing fraction of their indigenous software workers. Moreover, as noted by Kapur and McHale (2005b), the international market for software professionals is increasingly competitive. Richer countries such as the US, Canada, Australia, Germany, and the UK increasingly compete for

[31] Source: US Department of Homeland Security, Bureau of Citizenship and Immigration Services, adminstrative data.

[32] These data include migration of Irish citizens that have returned after studying in US universities, including those studying for computer science degrees.

talent from other countries. In many cases, this competition has manifested itself as a decline in the traditional barriers to short-term and long-term migration (Kapur and McHale 2005b). This competition is likely only to increase with the aging demographics of these countries as well as the increasing requirements for a skilled workforce in software and in other industries.

Federal Government Spending on Software R&D

US Federal government investment in computer hardware and software R&D is thought to be one of the contributing success factors to both industries (Flamm 1988; Langlois and Mowery 1996). Early government R&D investment in software provided the computer facilities for universities to conduct early software research (Langlois and Mowery 1996) and federal agencies such as NASA and ARPA have been long-standing supporters of computer-related research. Federal grants remain a major source of funding for doctoral students in computer science: in 2003, 17.4% of full time computer science graduate students reported that their primary source of funding was from the federal government. [33]

In the 1990s, though funding from the Department of Defense had largely flattened out, R&D spending grew rapidly throughout the decade through expanded funding from agencies such as the Department of Energy and NSF. However, over the period 2001-2003 (the most recent data available) government R&D spending in computer science has remained largely flat. Moreover, the percent of total R&D spending on computer science (relative to other fields) has declined over the period 2001-2003, from 4.5 percent to 4.0 percent.[34] We discuss the implications of these spending patterns in the next section.

7. Conclusions and Implications

[33] Source: National Science Foundation, Division of Science Resource Statistics, Survey of Graduate Students and Postdoctorates in Science and Engineering, WebCASPAR database (Science and Engineering Indicators 2006).
[34] Source: Science and Engineering Indicators (2006).

Public Policy Implications

The trends that we have described in this paper have several public policy implications. First, our results have provided evidence of a sizable export-driven software services sector in countries like India and Ireland, though there is less evidence of substantial inventive activity in software going on outside of the US. These results suggest that entry and mid-level programming jobs can be performed away from the point of final demand, though inventive activity that requires proximity with lead users is most effectively done in the US. However, these entry and mid-level programming jobs have traditionally provided US IT workers with the skills needed to perform more complicated development activities such as creation of new software programs (Levy and Murname 2004). In other words, training by US firms has traditionally bestowed a beneficial externality upon entry-level workers by providing them with general human capital that workers appropriate later in their careers. This human capital is not easily provided by traditional publicly funded primary or secondary school education programs (Levy and Murname 2004). As a result, declining demand for entry-level programming jobs could negatively impact US workers' future ability to perform more complex software development activity (e.g., new packaged software development).

If this is true, then there are two ways that US workers could obtain the general human capital needed. One would be for US workers to internalize the externality by accepting jobs for lower salaries. Of course, in the short run workers may prefer instead to accept jobs in other (relatively higher-paying) fields. Alternatively, government could attempt to subsidize entry-level employment by, for example raising the costs of H1-B visas or by direct labor market subsidies. However, if the cost of remote software development remains lower than that in the US, then clearly implementation of this policy may be problematic.

We have provided evidence of recent declines in computer science enrollments at the graduate and undergraduate levels. In our view, it is too soon to speculate whether these changes are evidence of a new trend or instead reflect temporary student reactions to business cycle fluctuations; in particular, the IT downturn that began in the early part of this decade. Still, there is evidence that for some time US software developers have been using skilled labor from abroad as inputs into their innovation production function, presumably in part to supplement the pool of skilled labor available locally. As noted above, there is increasing competition from other industrialized countries for these skilled workers, and there is no sign that this competition will abate in the near future. Decreasing the costs of H1-B visas or lowering the costs of permanent migration is unlikely to be feasible in the short run because of concerns of labor substitution between foreign and indigenous workers describe above. As a result, ensuring an adequate supply of local workers with sufficient basic or enabling skills (Levy and Murname 2004) in mathematics, computer science, and related fields taught in the nation's school and university system will be important to the long run success of software producers in the US.

Another area of public policy concern is in government funding of computer science research. As noted above, federal funding of computer science has flattened out in recent years. A continuation of this trend could negatively impact innovative activity in software in the US in two ways: by decreasing an importance source of financial capital for basic research, as well as potentially accentuating the negative downturn in enrollments in computer science graduate programs in the US through a decline in graduate student funding.

Summary and Conclusions

There are currently two very different stories in the globalization of software development. On the one hand, the IT services industries in countries such as India, Ireland, and other countries continue to grow rapidly. The production of IT services is quite dispersed globally, and this dispersion will only increase over time. In contrast, both sales and inventive activity in packaged software are localized in the US and undertaken primarily by US firms. There is no sign of these trends reversing in the short to medium run.

Recent trends in computer science enrollments have attracted considerable attention in the popular press. We do find evidence of some declines in enrollments in US computer science in recent years.[35] However, of likely equal or greater importance in the short run may be the increasing incentives for skilled foreign workers to remain in their home countries or to depart from the US immediately or some years after degree conferral. There is already some evidence that improving educational systems and employment opportunities in the underdog countries is causing some skilled software professionals to remain at home or to return.

Nonetheless, there are powerful forces at work that are likely to keep the development of new software products and software innovation concentrated in the US for some time to come. Despite recent trends, the US continues to have the best post-secondary educational systems for training computer scientists in the world, and it continues to enjoy substantial albeit declining inward migration that benefits the software (and other) industries. Beyond the education and human capital issues, US software innovators continue to enjoy substantial advantages due to agglomeration economies arising from the pre-existing concentration of the industry, as well as a generally favorable business environment. Perhaps the most significant advantage that US software product innovators enjoy is proximity to lead users. US firms have been among the most innovative users of IT in the world, and these users have benefited US software producers in the past and will continue to do so for some time to come.

[35] In graduate programs these declines appear to be concentrated primarily among immigrants. Among undergraduate degree programs current data are not available to indicate whether these declines are from US nationals or immigrants.

References

Allison, John and Emerson Tiller. 2003. Internet Business Method Patents. In *Patents in the Knowledge-Based Economy* (eds.) Wesley M. Cohen and Steve A. Merrill, p. 259-257, Washington: National Academies Press.

Allison, John, Arti Rai, and Bhaven N. Sampat. 2005. University Software Ownership: Trends, Determinants, Issues. Working Paper, Columbia University.

Anchordoguy, Marie. 2000. Japan's software industry: a failure of institutions? *Research Policy* 29: 391-408.

Armstrong, D.J. and P. Cole. 2002. Managing distances and differences in geographically distributed work groups. In *Distributed Work* (eds.) Pamela J. Hinds and Sara Kiesler, p. 167-186, Cambridge: MIT Press.

Arora, Ashish, V.S. Arunachalam, V.S. Asundi, and R. Fernandes. 2001. The Indian software services industry. *Research Policy* 30(8): 1267-1287.

Arora, Ashish and Surendrakumar Bagde. 2006. The Indian Software Industry: the Human Capital Story. Working Paper, Heinz School of Public Policy & Management, Carnegie Mellon University.

Arora, Ashish and Chris Forman. 2006. Proximity and Software Programming: IT Outsourcing and the Local Market. Forthcoming, *Proceedings of the 40th Annual Hawaii International Conference in System Sciences*.

Arora, Ashish and Alfonso Gambardella. 2004. The Globalization of the Software Industry: Perspectives and Opportunities for Developed and Developing Countries. NBER Working Paper #10538.

Arora, Ashish, Alfonso Gambardella, and Salvatore Torrisi. 2004. In the Footsteps of Silicon Valley? Indian and Irish Software in the International Division of Labor. In *Building High-Tech Clusters: Silicon Valley and Beyond* (eds.) Timothy Bresnahan and Alfonso Gambardella, p. 78-120, Cambridge: Cambridge University Press.

Aspray, William, Frank Mayadas, and Moshe Y. Vardi. 2006. Globalization and Offshoring of Software: A Report of the ACM Job Migration Task Force.

Athreye, Suma. 2005a. The Indian Software Industry. In *From Underdogs to Tigers: The Rise and Growth of the Software Industry in Brazil, China, India, Ireland, and Israel* (eds.) Ashish Arora and Alfonso Gambardella, p. 7-40, Oxford: Oxford University Press.

Athreye, Suma. 2005b. The Indian Software industry and its evolving service capability. *Industrial and Corporate Change* 14(3): 393-418.

Baba, Yasunori, Shinji Takai, and Yuji Mizuta. 1996. The User-Driven Evolution of the Japanese Software Industry: The Case of Customized Software for Mainframes. In *The International Computer Software Industry* (ed.) David C. Mowery, p. 104-130. Oxford: Oxford University Press.

Bessen, James and Robert M. Hunt. 2004. An Empirical Look at Software Patents. Working Paper 03-17/R, Research on Innovation.

Botelho, Antonio J. Junqueria, Giancarlo Stefanuto, and Francisco Veloso. 2005. The Brazilian Software Industry. In *From Underdogs to Tigers: The Rise and Growth of the Software Industry in Brazil, China, India, Ireland, and Israel* (eds.) Ashish Arora and Alfonso Gambardella, p. 99-130, Oxford: Oxford University Press.

Bresnahan, Timothy, and Shane Greenstein. 1996. Technical Progress in Computing and in the Uses of Computers. *Brookings Papers on Economic Activity, Microeconomics*: 1-78.

Bresnitz, Dan. 2005. The Israeli Software Industry. In *From Underdogs to Tigers: The Rise and Growth of the Software Industry in Brazil, China, India, Ireland, and Israel* (eds.) Ashish Arora and Alfonso Gambardella, p. 72-99, Oxford: Oxford University Press.

Brooks, Frederick. 1995. *The Mythical Man-Month: Essays on Software Engineering, 20th Anniversary Edition.* Addison-Wesley.

Copeland, Duncan G. and James L. McKenney. 1988. Airline Reservations Systems: Lessons from History. *MIS Quarterly* 12(3): 353-370.

Cottrell, Thomas. 1996. Standards and the Arrested Development of Japan's Microcomputer Software Industry. In *The International Computer Software Industry* (ed.) David C. Mowery, p. 131-164. Oxford: Oxford University Press.

European Patent Office. 2005. Computer-Implemented Inventions and Patents. Law and Practice at the European Patent Office.

Flamm, Kenneth. 1988. *Creating the Computer: Government, Industry, and High Technology.* Washington, DC: Brookings Institution Press.

Fransman, M. 1995. *Japan's Computer and Communications Industry: The Evolution of Industrial Giants and Global Competitiveness.* New York: Oxford University Press.

Furman, Jeffrey L., Michael E. Porter, and Scott Stern. 2002. The determinants of national innovative capacity. *Research Policy* 31(6): 899-933.

Campbell-Kelly. 2003. *From Airline Reservations to Sonic the Hedgehog: A History of the Software Industry*. Cambridge: MIT Press.

General Accounting Office. 2005. U.S. and India Data on Offshoring Show Significant Differences. GAO Report GAO-06-116.

Gereffi, Gary and Vivek Wadhwa. 2005. Framing the Engineering Outsourcing Debate: Placing the United States on a Level Playing Field with China and India. Report, Master of Engineering Program, Duke University.

Gompers, Paul and Josh Lerner. 2006. *The Venture Capital Cycle: Second Edition*. Cambridge: MIT Press.

Gormely, J., W. Blustein, J. Gatloff, and H. Chun. 1998. The runaway costs of packaged applications. *The Forrester Report* 3(5).

Graham, Stuart J.H. and David C. Mowery. 2003. Intellectual Property Protection in the U.S. Software Industry. In *Patents in the Knowledge-Based Economy* (eds.) W.M. Cohen and S.A. Merrill, p. 219-258. Washington, DC: National Academies Press.

Graham, Stuart J.H. and David C. Mowery. 2005. Software Patents: Good News or Bad News. In *Intellectual Property Rights in Frontier Industries: Software and Biotechnology* (ed.) Robert W. Hahn, p.45-80. Washington, DC: AEI-Brookings Joint Center for Regulatory Studies.

Griliches, Zvi. 1990. Patent Statistics as Economic Indicators. *Journal of Economic Literature* 28(4): 1661-1707.

Hall, Bronwyn H. and Megan MacGarvie. 2006. The Private Value of Software Patents. NBER Working Paper #12195.

Herbsleb, James D., Daniel J. Paulish, and Matthew Bass. 2005. Global Software Development at Siemens: Experience from Nine Projects. *International Conference on Software Engineering* p. 524-533. St. Louis, MO.

Ilavarasan, P. Vigneswara. 2006. R&D in Indian Software Industry. In *Managing Industrial Research Effectively* (ed. Roli Varma), p.134-143. ICFAI University Press.

Jaffe, Adam and Manuel Trajtenberg. 2002. *Patents, Citations, and Innovations: A Window on the Knowledge Economy.* Cambridge: MIT Press.

Kapur, Davesh and John McHale. 2005a. Sojourns and Software: Internationally Mobile Human Capital and High-Tech Industry Development in India, Ireland, and Israel. In *From Underdogs to Tigers: The Rise and Growth of the Software Industry in Brazil, China, India, Ireland, and Israel* (eds.) Ashish Arora and Alfonso Gambardella, p. 236-274. Oxford: Oxford University Press.

Kapur, Davesh and John McHale. 2005b. *Give Us Your Best and Brightest: The Global Hunt for Talent and its Impact on the Developing World*. Washington, DC: Center for Global Development.

Kraut, Robert E. and L.A. Streeter. 1995. Coordination in Software Development. *Communications of the ACM* 38(3): 69-81.

Langlois, Richard N. and David C. Mowery. 1996. The Federal Government Role in the Development of the U.S. Software Industry. In *The International Computer Software Industry* (ed.) David C. Mowery, p. 53-85. Oxford: Oxford University Press.

Layne-Farrar, Anne. 2005. Defining Software Patents: A Research Field Guide. Working Paper 05-14, AEI-Brookings Joint Center for Regulatory Studies.

Levy, Frank and Richard J. Murname. *The New Division of Labor: How Computers are Creating the Next Job Market.* New York: Russell Sage Foundation.

Malerba, Franco and Salvatore Torrisi. 1996. The Dynamics of Market Structure and Innovation in the Western European Software Industry. In *The International Computer Software Industry* (ed.) David C. Mowery, p. 165-196. Oxford: Oxford University Press.

March, James G. and Herbert A. Simon. 1958. *Organizations*. New York: Wiley.

Merges, Robert P. 2006. Patents, Entry, and Growth in the Software Industry. Working Paper, University of California, Berkeley.

Messerschmitt, David G. and Clemens Szyperski. 2002. *Software Ecosystem: Understanding an Indispensible Technology and Industry*. Cambridge: MIT Press.

Mokyr, Joel. 2002. *The Lever of Riches.* Oxford, UK: Oxford University Press.

Mowery, David C. and Nathan Rosenberg. 1998. *Paths of Innovation.* Cambridge, UK: Cambridge University Press.

Mullick, N., M. Bass, Z. Houda, R. Sangwan, D. Paulish, M. Cataldo, J. Herbsleb, L. Bass. 2006. Siemens Global Studio Project: Experiences adopting an integrated GSD infrastructure. Forthcoming, *Proceedings of the International Conference on Global Software Engineering.*

National Science Foundation. 2006. *Science and Engineering Indicators 2006*. Washington, DC.

Nelson, R. and R. Rosenberg. 1994. American Universities and Technical Advance in Industry *Research Policy* 23(3): 323-348.

Olson, G.M. and J.S. Olson. 2000. Distance Matters. *Human-Computer Interaction* 15(2 & 3): 139-178.

OECD. 2002. *OECD Information Technology Outlook*. Paris : OECD.

OECD. 2004. *OECD Information Technology Outlook*. Paris: OECD.

OECD. 2006. *OECD Information Technology Outlook.* Paris: OECD.

O'Riain, S. 1999. Remaking the Developmental State : The Irish Software Industry in the Global Economy. Doctoral Dissertation, Department of Sociology, University of California, Berkeley.

Parker, R.W. 1965. The SABRE System. *Datamation.* September: 49-52.

Parthasarathy, Balaji and Yuko Aoyama. 2006. From Software Services to R&D Services: Local Entrepreneurship in the Software Industry in Bangalore, India. *Environment and Planning* A 38(7): 1269-1285

Porter, Michael E. 1990. *The Competitive Advantage of Nations.* New York: Free Press.

Ricadela, Aaron. 2005. Q&A: Bill Gates on Supercomputing, Software in Science, and More. *Information Week.* November 18, 2005.

Romer, P. 1990. Endogenous Technological Change. *Journal of Political Economy* 98: S71-S102.

Rosenberg, Nathan. 1963. Technological Change in the Machine Tool Industry, 1840-1910. *Journal of Economic History* 23: 414-443.

Rosenberg, Nathan. 1983. Inside the Black Box: Technology and Economics. Cambridge: Cambridge University Press.

Sands, Anita. 2005. The Irish Software Industry. In *From Underdogs to Tigers: The Rise and Growth of the Software Industry in Brazil, China, India, Ireland, and Israel* (eds.) Ashish Arora and Alfonso Gambardella, p. 41-71. Oxford: Oxford University Press.

Scaffidi, Christopher, Mary Shaw, and Brad Myers. 2005. Estimating the Numbers of End Users and End User Programmers. *Proceedings of the 2005 IEEE Symposium on Visual Languages and Human-Centric Computing:* p. 207-214.

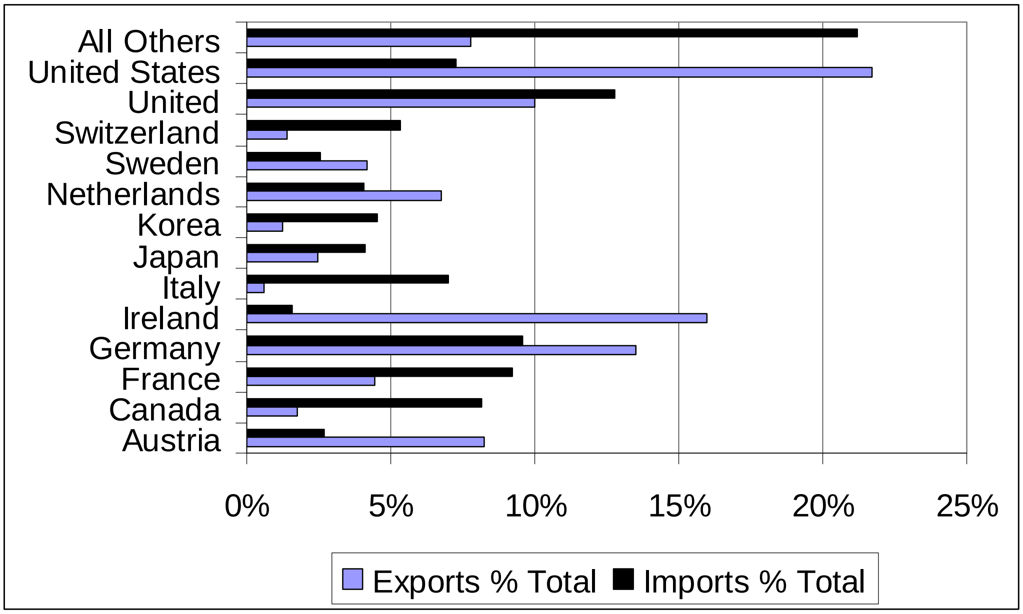
Schumpeter, Joseph A. 1934. *The Theory of Capitalist Development.* Cambridge, MA: Harvard University Press.

Steinmueller, Edward. 2004. The European software sectoral system of innovation. In *Sectoral Systems of Innovation: Concepts, Issues, and Analyses of Six Major Sectors in Europe* (ed.) Franco Malerba, p. 193-242. Cambridge: Cambridge University Press.

Thoma, Grid and Salvatore Torrisi. 2006. The Evolution of the Software Industry in Europe. Working Paper, CESPRI, Bocconi University.

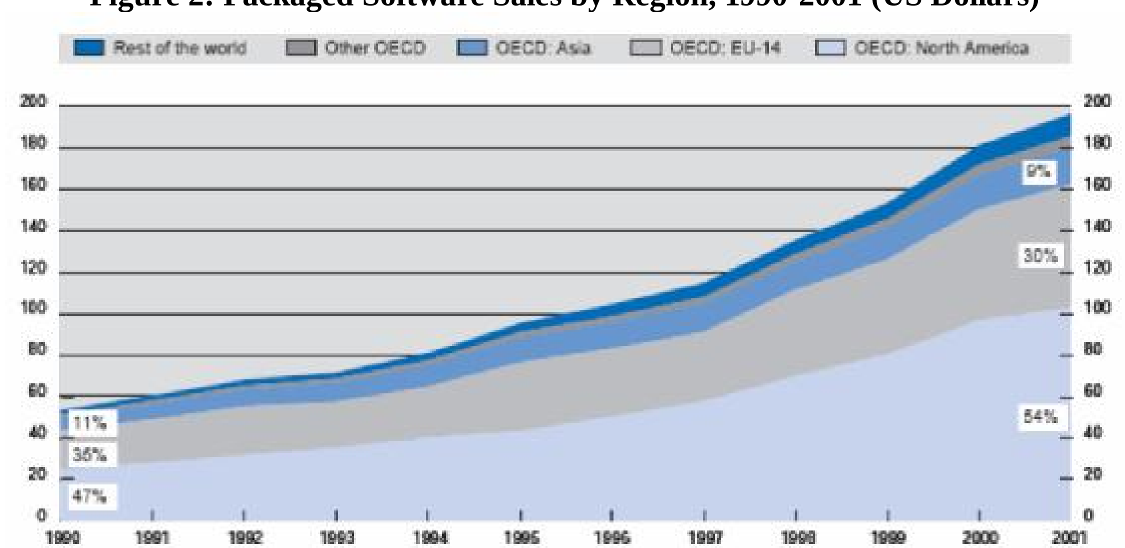
Tschang, Ted and Lan Xue. 2005. The Chinese Software Industry. In *From Underdogs to Tigers: The Rise and Growth of the Software Industry in Brazil, China, India, Ireland, and Israel* (eds.) Ashish Arora and Alfonso Gambardella, p. 171-206, Oxford: Oxford University Press.

Figure 1: Percentage of Total 2002 Software Product Exports and Imports by OECD Country



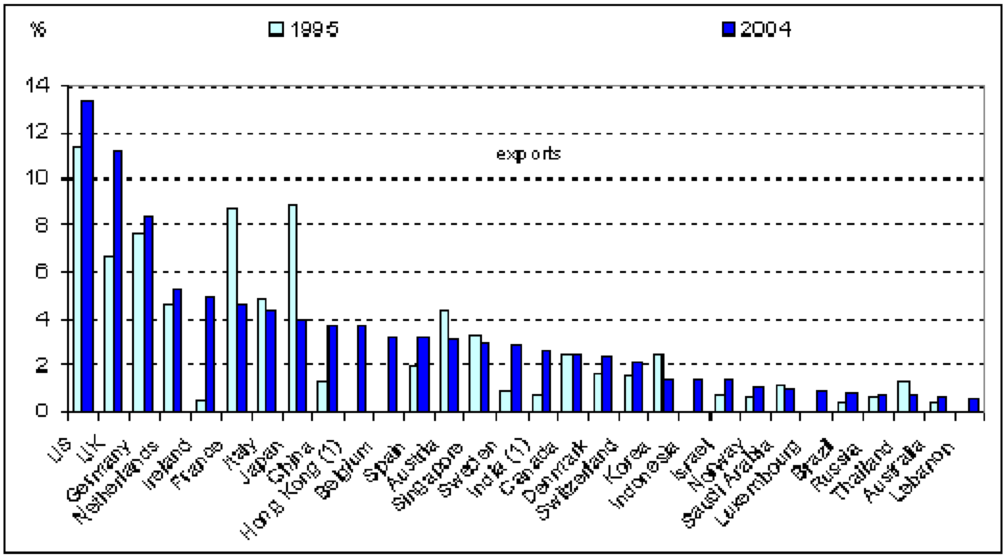
Source: OECD Information Technology Outlook 2004, Table C.1.8: OECD trade in software goods, 1996-2002. Compiled from International Trade Statistics database.

Figure 2: Packaged Software Sales by Region, 1990-2001 (US Dollars)



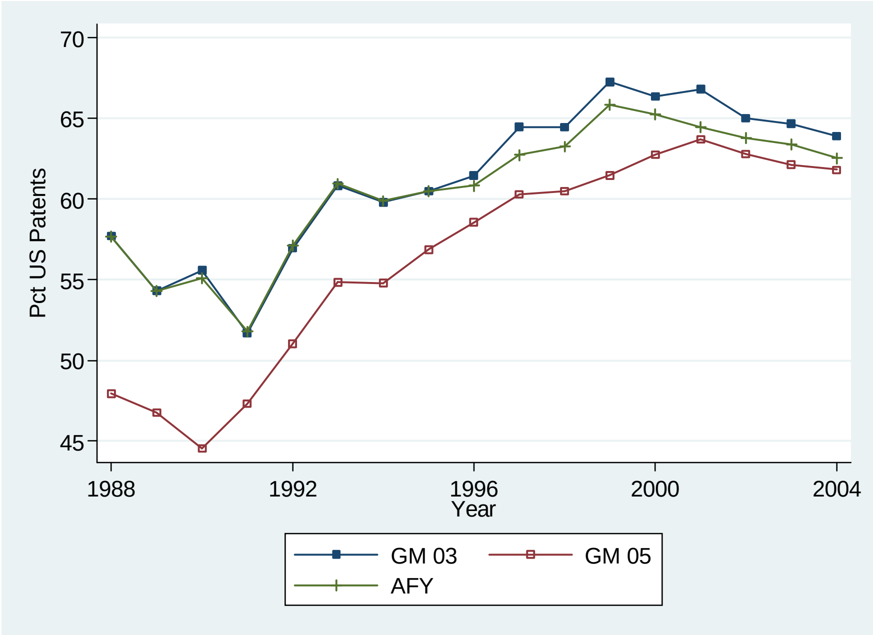
Source: OECD (2002) using IDC data. Reported in Thoma and Torrisi (2006).

Figure 3: Top 30 country shares of reported exports of other business services and computer and information services, 1995 and 2004



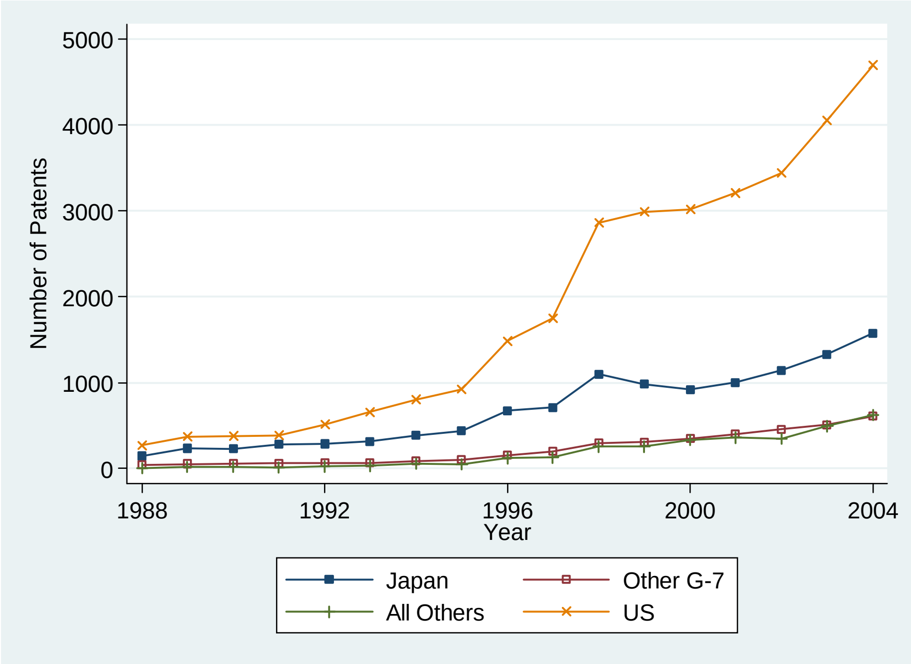
2004 data not yet available for all countries. For Hong Kong (China), India and the Slovak Republic data for 2003. Republished with permission from 2006 OECD Economic Outlook. Based on IMF Balance of Payments Database, March 2006.

Figure 4
Percent of US Patents Invented in US Under Different Software Definitions



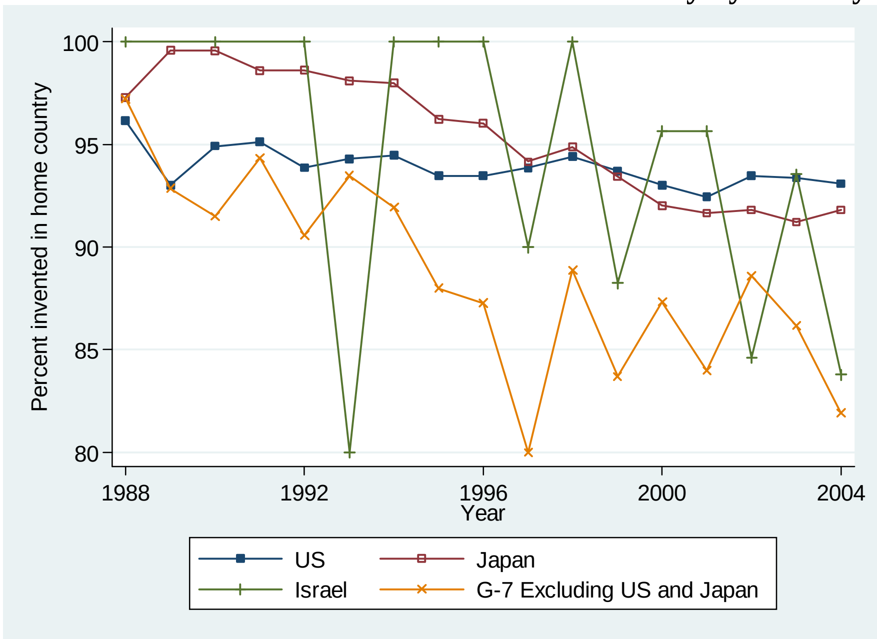
Source: USPTO data and authors' calculations.

Figure 5
US Software Patents invented in US and Other Countries



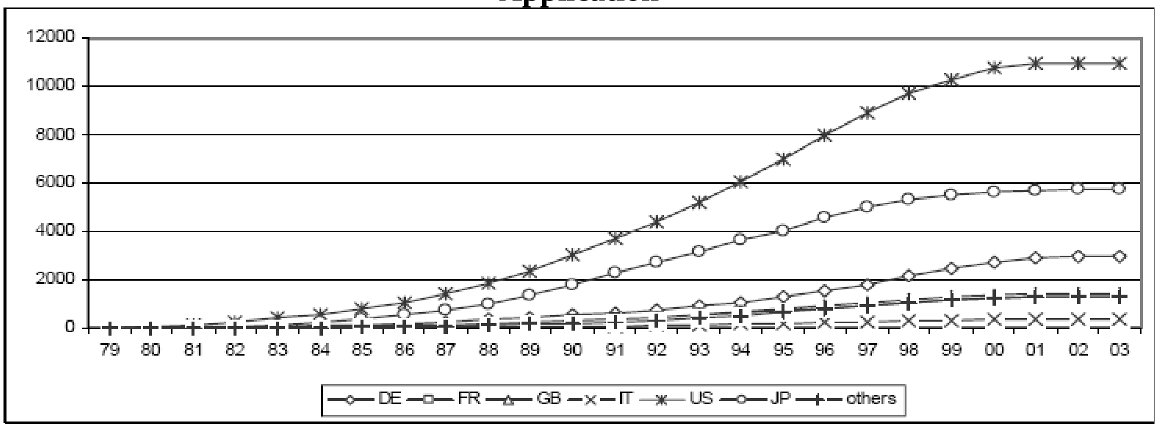
Source: USPTO data and authors' calculations.

Figure 6
Percentage of US Software Patents Invented in Home Country by Country of Assignee



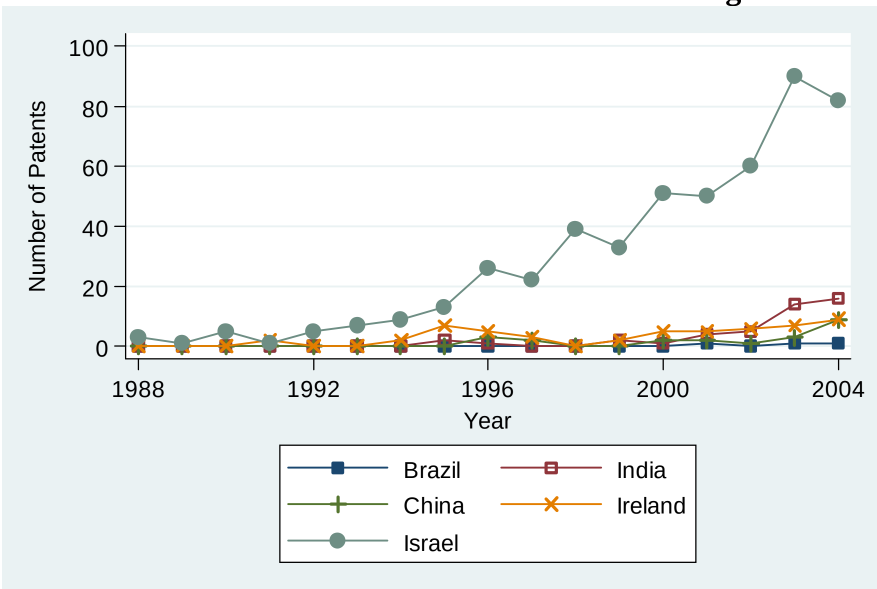
Source: USPTO data and authors' calculations.

Figure 7
European Patent Office Software Patent Grants by Country of the Assignee and Year of Application



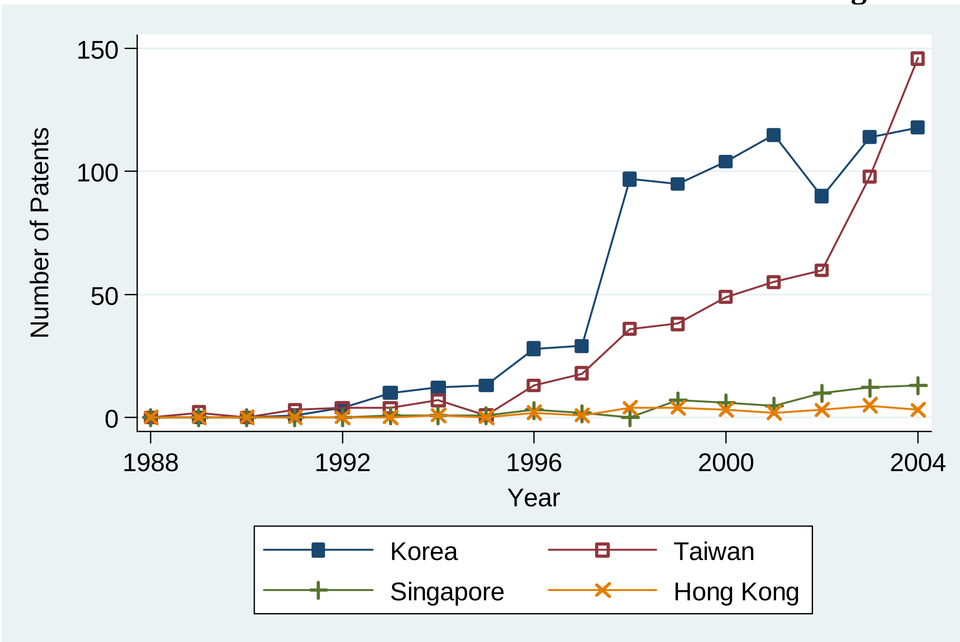
Source: Thoma and Torrisi (2006)

Figure 8
Number of US Software Patents Invented in Underdog Countries



Source: USPTO data and authors' calculations.

Figure 9
Number of US Software Patents Invented in East Asian Tigers



Source: USPTO data and authors' calculations.

Figure 10
Distribution of Assignee Country for US Software Patents Invented in Software Underdogs
(Top Panel: US; Bottom Panel: All Other Countries)

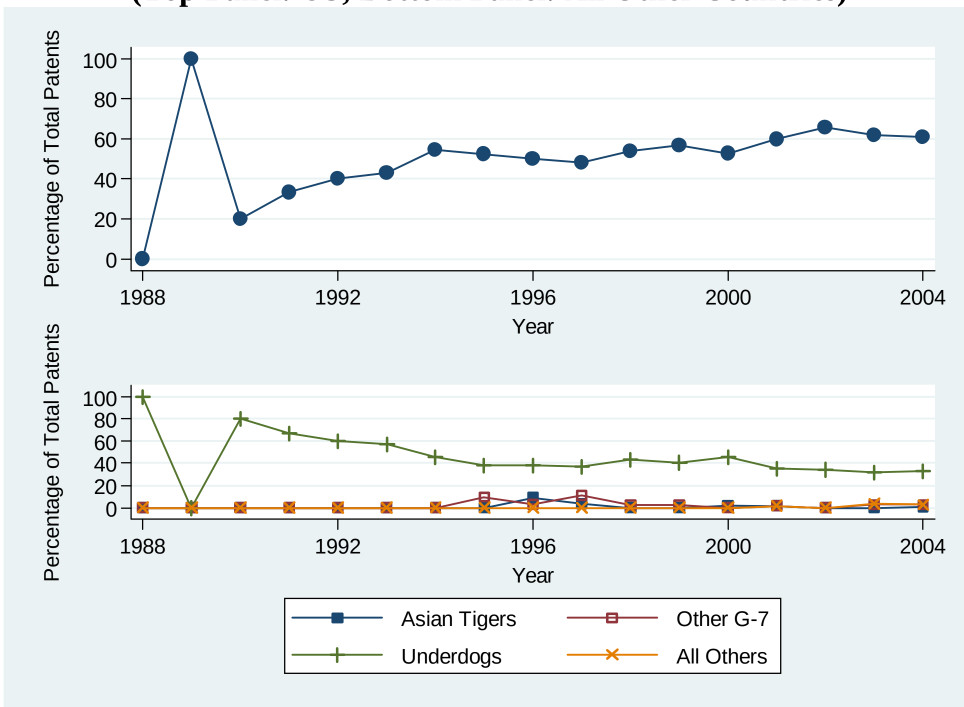
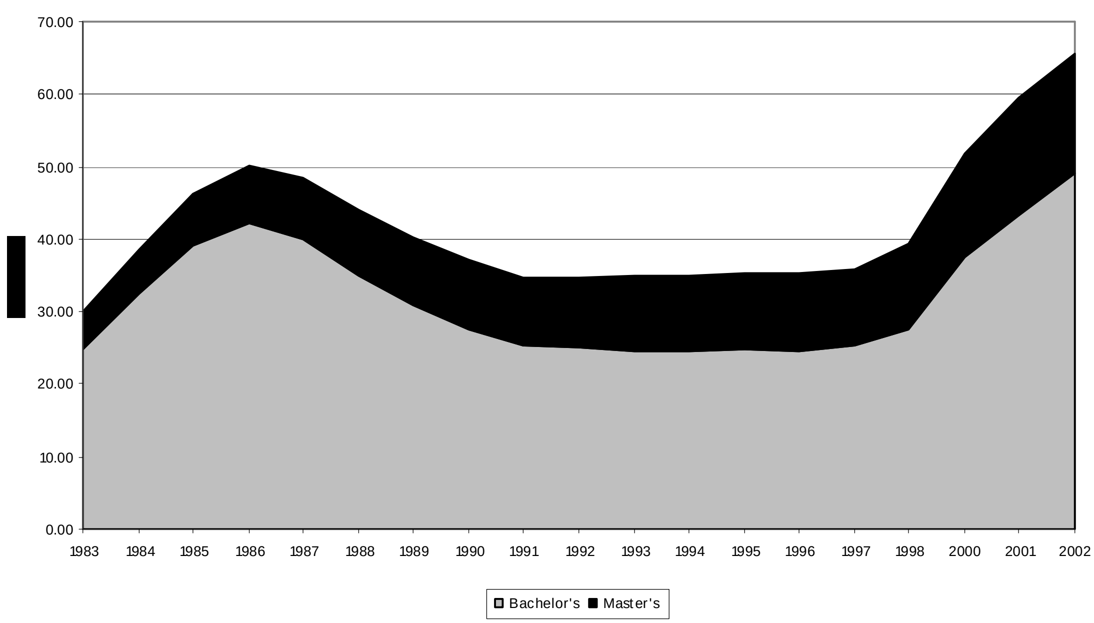
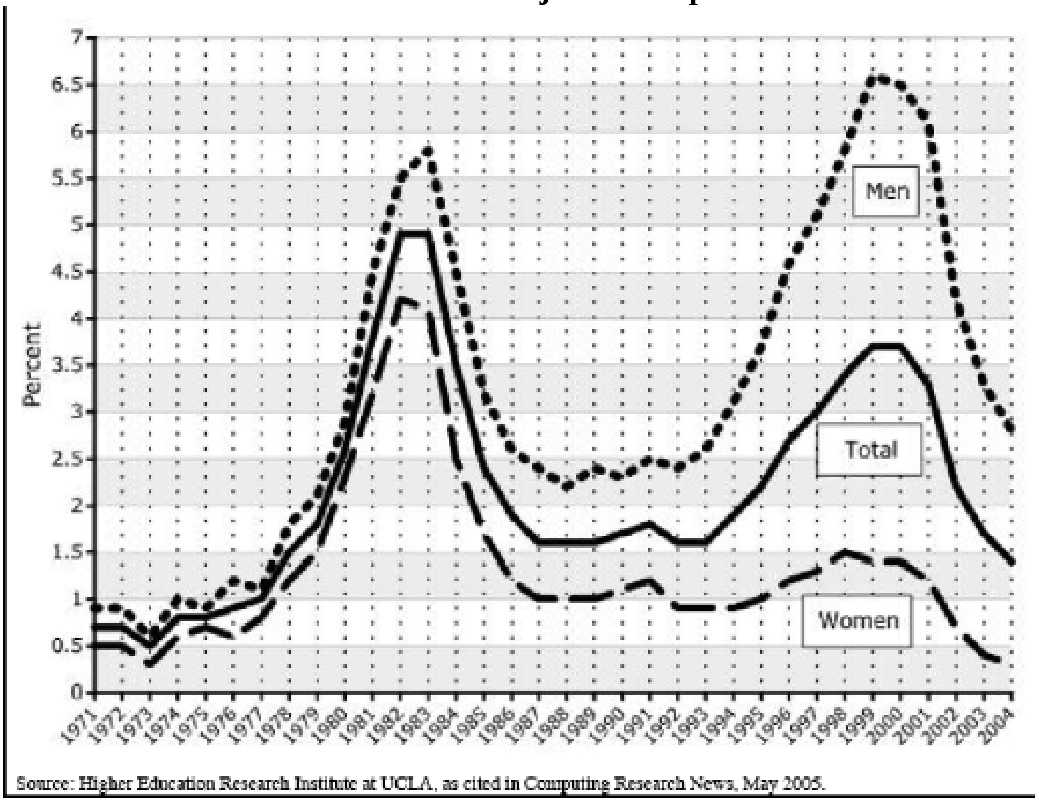


Figure 11
Undergraduate and Master's Degrees Earned in Computer Science



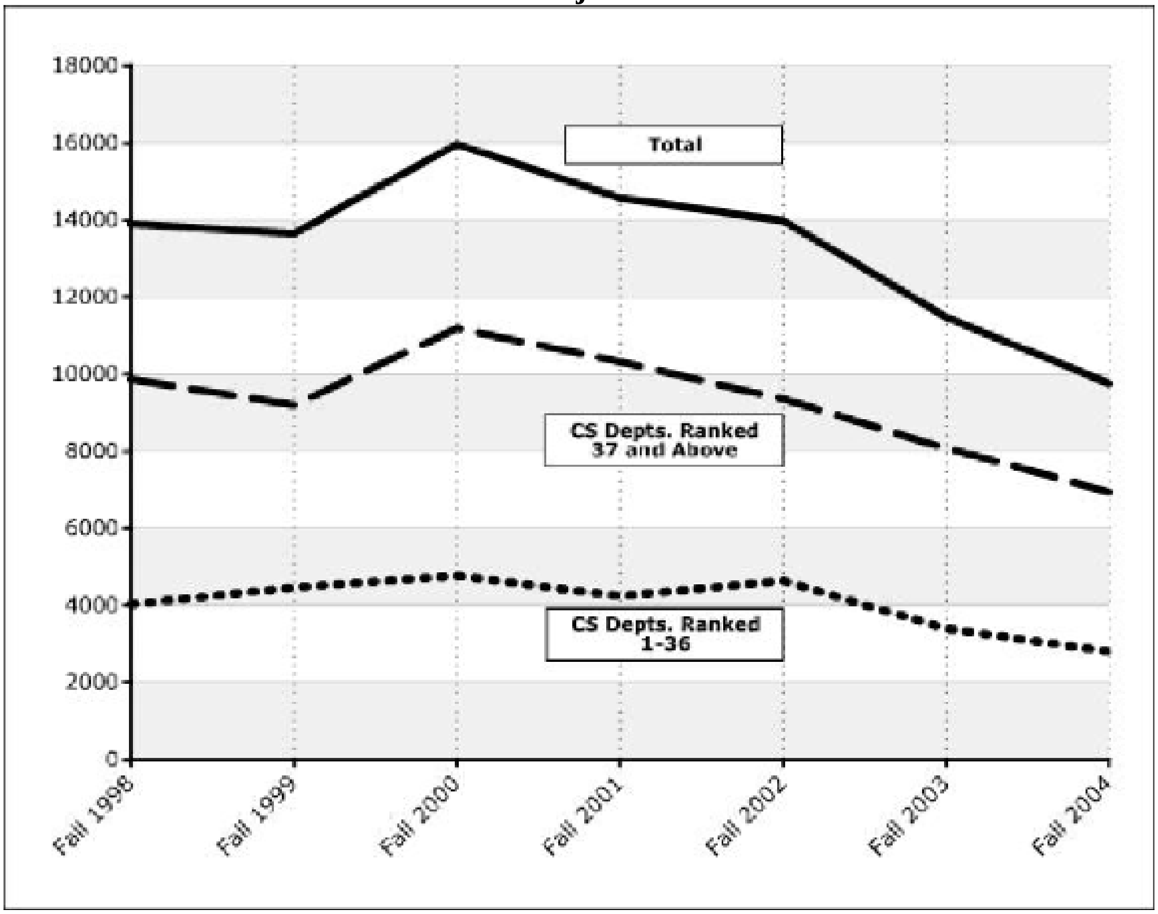
SOURCES: U.S. Department of Education, National Center for Education Statistics, Integrated Postsecondary Education Data System, Completions Survey; and National Science Foundation, Division of Science Resources Statistics, WebCASPAR database, http://webcaspar.nsf.gov. See appendix table 2-26. 1999 data is not available.

Figure 12
Freshman Intentions to Major in Computer Science



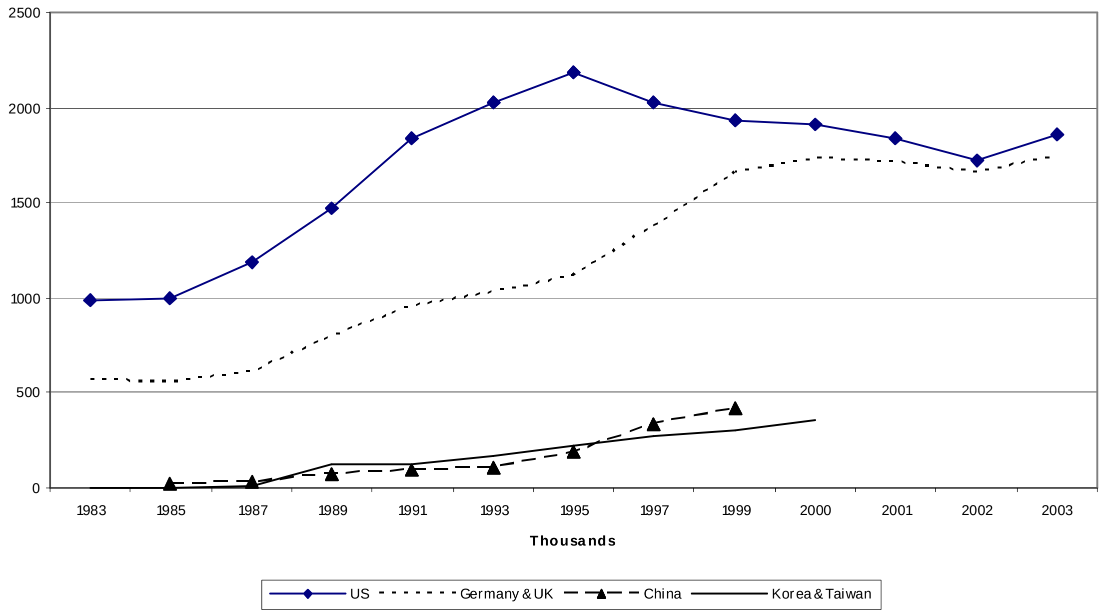
Source: Globalization and Offshoring of Software: A Report of the ACM Job Migration Task Force (2006).

Figure 13 Newly Declared Computer Science Majors



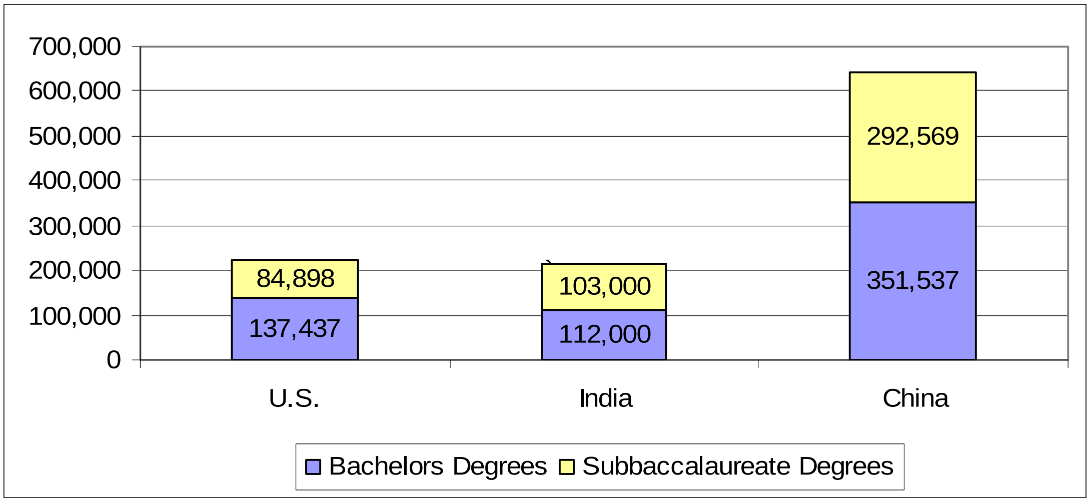
Source: Computing Research Association and Globalization and Offshoring of Software: A Report of the ACM Job Migration Task Force (2006).

Figure 14: Doctoral Degrees in Mathematics & Computer Science by Region



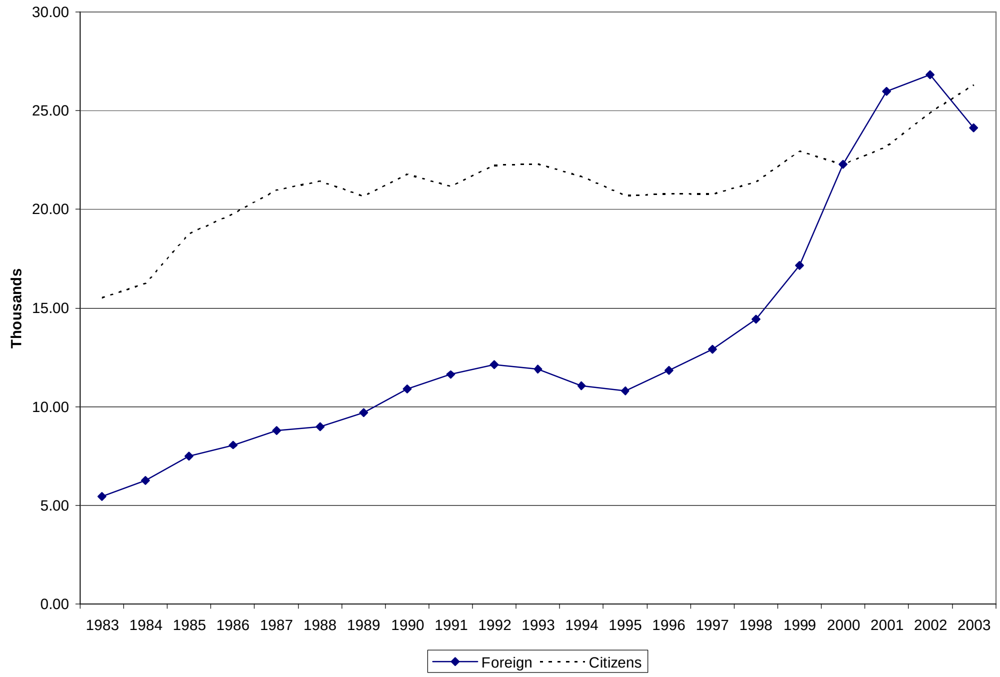
Sources: China—National Research Center for Science and Technology for Development and Educational Yearbook, 2002; Division of Higher Education, special tabulations (2005); South Korea—Organisation for Economic Co-operation and Development, Center for Education Research and Innovation, Education database, http://www1.oecd.org/scripts/cde/members/EDU_UOEAuthenticate.asp; and Taiwan—Ministry of Education, Educational Statistics of the Republic of China (annual series).

Figure 15
Bachelor's and Subbaccalaureate Degrees in Engineering, 2004



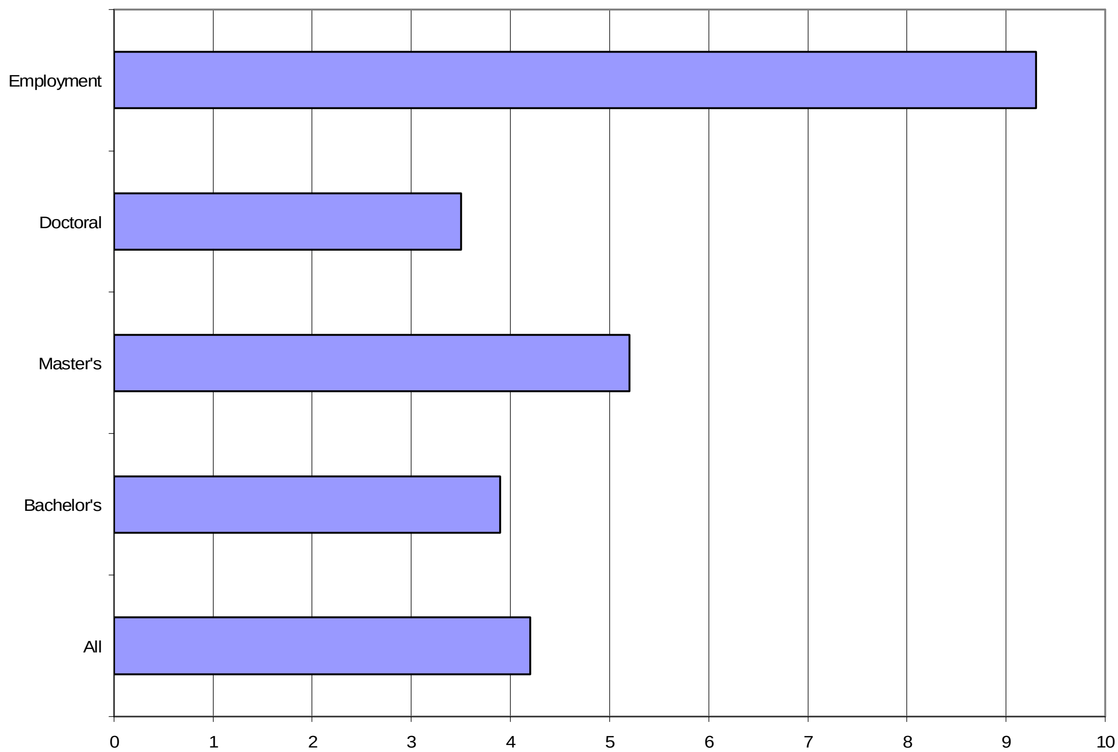
Source: Gereffi and Wadhwa (2005).

Figure 16
US Graduate Enrollment in Computer Science by Citizenship



Source: Science and Engineering Indicators (2006)

Figure 17
Average Annual Growth of Degree Production and Occupational Employment in Mathematics and Computer Science, 1980-2000



Source: Science and Engineering Indicators (2006)

Table 1. Computer and information services with Unaffiliated Foreigners (Mil Dollar)

Years	1994	1998			2004			AAGR, 1998-2004		
	Computer and Information Services	Computer and Information Services	Royalties and License Fees	Total	Computer and Information Services	Royalties and License Fees	Total	Computer and Information Services	Royalties and License Fees	Total
Exports										
All countries	2,332	3705	3195	6900	6,601	4261	10,862	10.10	4.92	7.86
Canada	333	430	125	555	1144	279	1423	17.71	14.32	16.99
Europe	899	1767	1508	3275	3281	1328	4609	10.87	-2.10	5.86
Japan	177	306	724	1030	327	1568	1895	1.11	13.75	10.70
Asian Tigers	117	200	…	…	163	…	…	-16.34	…	…
Underdogs										
Brazil	48	136	...	...	149	81	230	1.53	....	...
Israel	51	24	32	56	38	13	51	7.96	-13.94	-1.55
China	17	29	46	75	48	51	99	8.76	1.73	4.74
India	9	38	17	55	227	29	256	34.70	9.31	29.21
Imports										
All countries	286	1494	498	1992	2002	589	2591	5.00	2.84	4.48
Canada	34	589	9	598	1189	12	1201	12.42	4.91	12.32
Europe	122	259	449	708	400	562	962	7.51	3.81	5.24
Japan	20	41	26	67	15	1	16	-15.43	-41.90	-21.23
Asian Tigers	6	18	…	…	31	…	…	55.98	…	…
Underdogs										
Brazil	1	1	1	2	1	...	...	0.00	...	...
Israel	0	9	2	11	7	3	10	-4.10	6.99	-1.58
China	2	6	...	...	7	...	...	2.60	...	...
India	7	100	...	...	315	6	321	21.07	...	...

Source: BEA Data on U.S. International Trade in Services. Ommitted cells include either transactions below $500,000 or data that were omitted to maintain confidentiality. Cross-border exports to and imports from unaffiliated foreigners of computer and information services are shown in table 1. Computer and information services (NAICS 518) include "computer and data processing services (NAICS 5181)", and "database and other information services (NAICS 5182)". This table was reorganized based on the tables of ‘Business, Professional, and Technical Services with Unaffiliated Foreigners’ from BEA. Ireland is include in all other EU, and not identified in BEA’s tables. These export and import transactions with unaffiliated foreigners are interfirm transfers, which are traditional trades. Note that “affiliated foreigners” are locally established affiliates of multinational firms. The Asian Tigers consist of Korea, Singapore, Taiwan, and Hong Kong.

Table 2: Growth in Employment for Foreign Affiliates of US Firms vs. Growth for All US Establishments, Selected Industries, 1999-2002

	1999	2002	AAGR
Information Services and Data Processing Services			
Foreign Affiliates of US Firms	104.5	132.0	8.1
All US Establishments	371.9	473.8	8.4
Computer System Design and Related Services			
Foreign Affiliates of US Firms	157.9	172.9	3.1
All US Establishments	997.0	1061.3	2.1
Total			
Foreign Affiliates of US Firms	262.4	304.9	5.1
All US Establishments	1368.9	1535.1	3.9

Source: Data on Foreign Affiliates of US Firms from Table on Selected Data for Majority-Owned Nonbank Foreign Affiliates and Nonbank U.S. Parents in All Industries, 2003. From BEA International Economic Accounts, U.S. Direct Investment Abroad: Financial and Operating Data For U.S. Multinational Companies. Data on all US establishments from US County Business Patterns data.

Table 3: Disclosed Rounds of Venture Financing by Country, 1988-2005
(Thousands of dollars)

	United States	Other G-7	Underdogs	All Other	Total
1988	2,565	660	0	0	3,225
1989	15,000	2,465	0	0	17,465
1990	6,350	464	248	0	7,062
1991	1,100	0	0	0	1,100
1992	1,607	1,418	0	0	3,025
1993	15,247	582	0	0	15,829
1994	7,403	138	0	0	7,541
1995	14,340	0	0	0	14,340
1996	92,784	1,466	0	2,766	97,016
1997	242,873	0	0	7,049	249,922
1998	300,355	9,359	0	6,039	315,753
1999	1,068,310	68,011	28,666	21,102	1,186,089
2000	2,036,591	221,297	73,307	169,636	2,500,830
2001	460,911	83,944	32,256	16,629	593,740
2002	99,836	23,295	6,831	3,815	133,777
2003	173,205	14,607	15,251	167	203,230
2004	151,025	9,492	10,600	1,848	172,965
2005	138,428	2,000	2,000	59	142,487

Source: Venture Economics VentureXpert database, and author's calculations. Software includes rounds of financing from software and e-commerce software firms. Dates are round date of financing.

Table 4: Sales of Software Products and IT Services in the EU-15

	2003	2004	2005	Avg Growth
Software Products	59,235	61,707	64,979	4.74%
System Software	30,944	32,537	34,536	5.64%
Application Software	28,291	29,169	30,443	3.73%
IT Services	112,472	116,149	120,913	3.68%
Professional Services	81,376	84,380	88,147	4.08%
Support Services	31,096	31,769	32,766	2.65%
Total Software	171,707	177,856	185,892	
Pct Services	52.67%	53.13%	53.74%	

Source: European Information Technology Observatory (2006).

Table 5: List of IPC Patent Classes Used in Analyses

Class/Subclass	**Description**
G06F 3/00	Input arrangements for transferring data to be processed into a form capable of being handled by the computer; Output arrangements for transferring data from processing unit to output unit, e.g. interface arrangements
G06F 5/00	Methods or arrangements for data conversion without changing the order or content of the data handled
G06F 7/00	Methods or arrangements for processing data by operating upon the order or content of the data handled
G06F 9/00	Arrangements for program control, e.g. control unit
G06F 11/00	Error detection; Error correction; Monitoring
G06F 12/00	Accessing, addressing or allocating within memory systems or architectures
G06F 13/00	Interconnection of, or transfer of information or other signals between, memories, input/output devices or central processing units
G06F 15/00	Digital computers in general
G06F 17/00	Digital computing or data processing equipment or methods, specially adapted for specific functions
G06K 9/00	Methods or arrangements for reading or recognizing printed or written characters or for recognizing patterns, e.g. fingerprints
G06K 15/00	Arrangements for producing a permanent visual presentation of the output data
G06T 11/00	Two dimensional (2D) image generation, e.g. from a description to a bit-mapped image
G06T 15/00	Three dimensional (3D) image rendering, e.g. from a model to a bit-mapped image
G09G 5/00	Control arrangements or circuits for visual indicators common to cathode-ray tube indicators and other visual indicators
H04L 9/00	Arrangements for secret or secure communication

Class names are as follows: G06F: Electric Digital Data Processing; G06K: Recognition of Data; Presentation of Data; Records Carriers; Handing Record Carriers; G06T Image Data Processing or Generation, in General; G09G Arrangements or Circuits for Control of Indicating Devices Using Static Means to Present Variable Information; H04L: Electric Communication Technique. Source: International Patent Classification System, World Intellectual Property Organization, http://www.wipo.int/classifications/ipc/ipc8/?lang=en.

Table 6: Assignee Industry for US Patents by Region, 1988-2005

	Industrial Machinery (SIC 35)	Electrical & Electronic Equip (SIC 36)	Holding Companies (SIC 67)	Software Publishers (SIC 7372)	All Other
United States	9741 (27.48)	5291 (14.92)	4770 (13.45)	2217 (6.25)	13,434 (37.89)
Other G-7	1023 (5.87)	153 (0.88)	12,561 (72.11)	98 (0.56)	3585 (20.58)
Asian Tigers	47 (2.77)	51 (3.00)	781 (46.00)	1 (0.06)	818 (48.17)
Software Underdogs	133 (18.02)	136 (18.43)	56 (7.59)	59 (7.99)	354 (47.97)
All Other	85 (6.42)	44 (3.33)	609 (46.03)	15 (1.13)	570 (43.08)

Source: Author's manipulation of data from USPTO and Corptech database of Technology Companies. Numbers represent frequencies of row and column combinations. Numbers of parentheses represent the percentage of assignees in an industry conditional on invention in the country in the row. The unit of observation in the table is a patent.

Table 7: Leading Recipients of US Software Patents, by Country of Inventor, 1988-2005

Name	Year	Industry	# of Employees	Revenue	Home Country	Number of patents
CHINA						
Microsoft	1975	Software	71,553	$44 Billion	US	6
IBM	1888	IT hardware, software, services	330,000	$91 Billion	US	5
United Microelectronics Corp.	1980	Electronics	12,000		Taiwan	4
Intel	1968	Electronics	99,900	$39 Billion	US	2
Huawei Technologies	1988	Telecommunication	44,000	$8.2 Billion	China	1
GERMANY						
Siemens	1847	Conglomerate	472,000	$75 Billion	Germany	252
Robert Bosch GmbH	1886	Automotive	251,000	$55 Billion	Germany	178
IBM	1888	IT hardware, software, services	330,000	$91 Billion	US	98
Infineon Technologies	1999	Electronics	36,000	$7 Billion	Germany	41
Daimler Chrysler AG	1998	Automotive	383,000	$150 Billion	Germany	38
UNITED KINGDOM						
IBM	1888	IT hardware, software, services	330,000	$91 Billion	US	140
International Computers Limited	1968	Computers			UK	40
British Telecommunnications PLC	1846	Telecommunications	104,400	$37 Billion	UK	38
Sun Microsystems Inc.	1982	IT Hardware	31,000	$11 Billion	US	35
Philips Corporation	1891	Electronics	159,226	$36 Billion	The Netherlands	32
IRELAND						
3Com Corporation	1979	Networks	1,925	$800 Million	US	11
Analog Devices Inc.	1965	IT Hardware	8,800	$2.4 Billion	US	3
Richmount Computers Limited						3
Hitachi Ltd.	1920	IT hardware, electronics	323,072	$80.5 Billion	Japan	3

Name	Year	Industry	# of Employees	Revenue	Home Country	Number of patents
IBM	1888	IT hardware, software, services	330,000	$91 Billion	US	3
ISRAEL						
IBM	1888	IT hardware, software, services	330,000	$91 Billion	US	69
Intel	1968	Electronics	99,900	$39 Billion	US	58
Motorola Inc.	1928	Electronics	88,000	$37 Billion	US	32
Scitex Corporation (now Scailex Corporation)[1]		IT hardware, now venture capital		$128.2 million	Israel	12
Applied Materials Inc.	1967	Semiconductor	12,576	$7 Billion	US	11
INDIA						
IBM	1888	IT hardware, software, services	330,000	$91 Billion	US	17
Texas Instruments	1930	Hardware	30,300	$13 Billion	US	12
Honeywell International Inc.	1886	Aerospace	116,000	$26 Billion	US	3
Veritas Operating Corporation (acquired by Symantec)[3]	1989	Software	16,000	$4.1 Billion	US	3
Sun Microsystems Inc.	1982	Hardware	31,000	$11 Billion	US	2
JAPAN						
Hitachi Ltd.	1920	IT hardware, electronics	323,072	$80.5 Billion	Japan	1403
Canon	1937	Imaging	100,000	$35 Billion	Japan	1286
Fujitsu	1935	Hardware	158,000	$40 Billion	Japan	1127
NEC Corporation	1899	Electronics	148,540	$41 Billion	Japan	976
Toshiba	1904	Electronics	165,000	$60 Billion	Japan	820
SOUTH KOREA						
Samsung	1938	Electronics		$80 Billion	South Korea	460
LG Electronics [4]	1958	Electronics	66614	$23.5 Billion	South Korea	100
Electronics and Telecommunications Research Institute	1976				South Korea	55

Name	Year	Industry	# of Employees	Revenue	Home Country	Number of patents
Hyundai Electronics (now Hynix Semiconductor)		Semiconductors	13000	$5.6 Billion	South Korea	49
Hyundai Motor Company	1967	Automotive	51000	$57 Billion	South Korea	30
UNITED STATES						
IBM	1888	IT hardware, software, services	330,000	$91 Billion	US	4981
Intel	1968	Electronics	99,900	$39 Billion	US	1648
Microsoft	1975	Software	71,553	$44 Billion	US	1136
Sun Microsystems Inc.	1982	Hardware	31,000	$11 Billion	US	1088
Hewlett-Packard Inc.	1939	Hardware	150,000	$89 Billion	US	682

Source: The top five firms with the largest number of US patents, identified from our calculations of USPTO data. Company data is from Hoover’s Online, company annual reports, company web pages, and Wikipedia. Revenues are in US dollars and for the most current year available. Missing cells represent firms for which we were unable to recover data.

[1] Data are for Scailex.

[2] Data are for msystems Ltd.

[3] Data are for Symantec.

[4] LG Electronics and LG Semicon Co. Ltd. are each part of the LG Group. LG Phillips is a joint venture with the LG Group and Philips. Missing data are because data on some subsidiaries of the LG Group are not available separately.

[5] Digital Equipment Corporation was acquired by Compaq Computer Corporation, which was subsequently acquired by Hewlett-Packard. Data are for Hewlett-Packard.

Table 8: Average Outsourcing by Size of Metropolitan Statistical Area

	Programming	Programming & Design	Hosting Ex Internet
Rural Area	17.81%	24.30%	15.91%
	(0.38%)	(0.43%)	(0.37%)
Small MSA (< 250,000)	17.87%	23.85%	15.04%
	(0.54%)	(0.60%)	(0.50%)
Medium MSA (250,000 to 1 million)	18.48%	26.30%	16.41%
	(0.35%)	(0.40%)	(0.34%)
Large MSA (> 1 million)	18.54%	26.08%	15.31%
	(0.21%)	(0.24%)	(0.20%)

Note: Calculations for 2002. Standard errors in parentheses. Difference between rural/small and medium/large is sig at 5% level for all three types. Source: Arora and Forman (2006)

Table 9: Output of Engineering Graduates (BS and BE) in India, various years

Year	Total Number of Engineering Graduates Produced
1990	42022
1991	44281
1992	46762
1993	48281
1994	52905
1995	56181
1996	57193
1997	61353
1998	67548
1999	75030
2000	79343
2001	97942
2002	107720
2003	128432

Source: Arora and Bagde (2006)

Notes: These data are based on the figures for the 14 major states (except the State of Bihar) in India, which account for 80% of the GDP and likely more than that number of the total production of engineering graduates. These data are based on “Annual Technical Manpower Review (ATMR)” reports published by National Technical Manpower Information System NTMIS, India. These reports are prepared by a state-level nodal center of NTMIS and give details of sanctioned engineering college capacity and outturn for all undergraduate technical institutions in the state. See cited source for more details.