Press Briefing
October 17, 1996

Good Programmers Are Hard To Find:
An Alternative Perspective On The Immigration Of Engineers

Press contact: Leda Karabela, Communications Director
Stanford Computer Industry Project
415.725.6367, lkarabela@leland.stanford.edu

Technical contacts: Avron Barr and Shirley Tessler
Co-Directors, SCIP Software Industry Study
415.725.8594, scip@cs.stanford.edu

Are low-paid foreign programmers taking jobs away from American engineers? Or are they alleviating a critical skill shortage in the US software industry?

Economic output requires raw materials and skilled labor. Every industry, from textiles to telecommunications, requires a certain number of workers with considerable skills. A generation ago, software engineering was not on the top of the list of skills required in any industry. Today, software-related expertise may be the most important skill needed in dozens of industries.

Recently, the Software Industry Study at Stanford's Computer Industry Project (SCIP) has been looking at the worldwide supply of software talent. It appears that this resource may now be fully tapped, everywhere. As a result, US immigration policy should recognize the importance of the supply of good programmers to our national economy, since neither educational nor technological developments will address the growing shortage of good programmers in the near term.

There is already a significant shortage of some software skills in the US

There are at present thousands of unfilled positions for specific software skills ranging from SAP configuration specialist to Oracle consultant to webmaster. While these job openings might reflect technology "fads," we think they represent a deeper trend in

the software labor pool. Specifically, we think this is an early sign of a skill shortage that is likely to grow very rapidly. Since the US Department Labor does not collect detailed data on software professionals and software jobs, we are only beginning to gather evidence to support this assertion, but our conversations with industry insiders strongly confirm our suspicions of a growing shortfall of good programmers.

The origin of this shortage was masked for a few years by large-scale downsizing in corporate IS departments and in aerospace in the late '80's and early '90's. Tens of thousand of programmers were laid off and reabsorbed by the economy in just a few years, which masked the steady growth in demand for their skills. However, because of the continued double-digit growth of the software products and services market and the ever-increasing use of software-based product functionality in so many industries, the demand for programmers is likely to continue to increase steadily for years to come. Furthermore, the programs that will need to be written will continue to increase in complexity, requiring higher levels of software skills.

Some parts of the software industry, like Silicon Valley start-ups and the major software publishers, are beginning to see the first signs of a labor shortage: hiring delays, rising salaries, and increased mobility among software professionals. The increase in the number of programmers who go out on their own as contractors or "boutique" software services firms also indicates a seller's market for software talent.

In other important parts of the software development world, the "top half of the class" is no longer available at all, at any price. The best talent in each year's crop of newly graduating programmers are much less likely to head for a corporate MIS department than they are to join a software publisher, even if they don't start their own company right away. As a result, the Indian software export industry, for example, which started-out as a cost-savings alternative, has become a major source of talented programmers. In just 5 years, India has reached $500M in annual services exports by filling a growing skill shortage in MIS departments in the US, Japan and Europe.

The consequences of a shortage of skilled programmers may be costly, or deadly

While it has become a hugely successful industry in the US, all is not well in the world of software. Every year we are building increasingly more complex software, and requiring it to keep pace with our always-changing business strategies and product designs. It doesn't always work.

Last year we saw early signs of trouble in news stories about the "little bug" in the Pentium chip, the delayed opening of Denver 's new airport, and the cancellation of the California DMV's $40 million systems integration project. These seemingly unrelated incidents are the tip of an iceberg. A recent study by the Standish Group estimates in the US alone, canceled software projects cost $81B in 1995, and another $59B for projects over budget.

We build today's systems the same way we've always built software — we depend on highly-skilled engineers who type fast. The best software architects, systems designers and programmers are already hard at work all over the world, and the demand for software of all sorts is increasing more rapidly than ever. Some of this software runs

our communications and financial infrastructure. Some of it runs our automobiles and airplanes.

The worldwide shortage of talented software professionals of all sorts will be perceived as a major economic issue in the next few years. If software development techniques don't change and the demand for ever more complex software continues to escalate, we will soon begin to see the signs of a global talent shortage:

- The public will become increasingly aware of software systems failures, including failure to finish building many systems, and breakdowns of organizations and devices which are dependent on complex software systems. Eventually, these failures will cause loss of life.

- Worldwide competition for (and emigration of) the best software talent will intensify, and corresponding increases in salaries, which are not low now, should be expected.

- Calls for certification to help differentiate the competent engineers and qualified services firms will be heard, and we will see increased differentiation among types of software professionals and services.

- Attention to changing the way we develop software and the way we train future professionals will increase.

Japan has already identified software as a critical, distressed national industry. Many countries have focused on the software industry as an economic engine for their future. Worldwide, the public will become increasingly aware of this industry over the next few years as it continues its dramatic growth and increasingly influences so many aspects of our lives.

Talented programmers are already fully engaged all over the world and there are no reserves

Some studies show that the best programmers are 20 times better than average programmers, whether you measure speed, error rate, or overall output. There is a "talent" element in the labor resource for this industry — not everyone can become a good programmer. And since the talent that makes for a good programmer may very well be the same skills that characterize a good electrical engineer or a good diagnostician, the number a good programmers a nation produces is limited by raw talent and by competition for that talent from other possible occupations.

In India, the combination of an excellent engineering education system and the fact that there are few alternative careers for talented engineers created a giant pool of software people. Clever entrepreneurs and effective government policies turned that resource into an important export industry for India. However, although there are talented programmers all over the world, there are no other countries with the enormous and underutilized engineering talent pool, English competency, investment in infrastructure, and entrepreneurial energy that made India's software industry such a success. We believe there will be no more Indias in the next 10 years. There are no ready pools of good programmers who can fill the gap.

No technological solutions are appearing on the immediate horizon either. We know of no technology that will adequately increase the productivity and effectiveness of current or next-generation software engineers. Object technology and the notion of assembling software from off-the-shelf components hold promise, but will require a generation to have widespread impact. Research on new ways of building large systems, like Prof. Gio Wiederhold's work on "mediation" technology at Stanford, may impact on the way programs are designed and developed in the longer term.

Can we educate a new cadre of talented software people? Degree programs in software engineering and other measures to train thousands of new programmers with the needed analysis, design, development, and project management skills are only beginning to be mentioned in curriculum discussions in the university, junior college, or post- graduate education systems. For example, the UC Extension produced a curriculum of this nature just this year. Regional software industry associations in Massachusetts, Minnesota, Silicon Valley and elsewhere have also begun to work with corporations and educational institutions to address local shortfalls in skilled manpower. Some organizations, like the newly-formed National Software Council, have called for certification programs to establish standard levels of professional competence, but these efforts are barely begun and not likely to be embraced quickly by the notoriously individualistic software community.

Immigration policy must continue to allow exceptions for talented software professionals

There is a growing movement in the US to cut down on immigration, both legal and illegal. Several engineering professional organizations, including the American Electronics Association, have pointed to immigrant engineers with lower wages as a threat to American jobs. We are concerned that proposed immigration policy changes, if applied without exceptions for software professionals, would cut the flow of the world's best programmers to the States. These measures are unnecessary, since there is a shortage of skilled labor in software, and could be damaging to the software industry and to the economy generally. The US is still the best place for good programmers to work, and we should encourage them to come here while they still have fewer domestic alternatives.

––––––––––

Avron Barr and Shirley Tessler are consultants to SCIP and co-directors the Software Industry Study. Mr. Barr was an early pioneer of expert systems technology and editor of the four-volume *Handbook of Artificial Intelligence.* He was a founder of Teknowledge and has, since 1981, been a management consultant to advanced software technology companies and their customers. Ms. Tessler is a Wharton MBA and Stanford MS/CS with 15 years' experience in banking and corporate finance. Her expertise is in software startup financing and management. Her current research interests include electronic commerce and electronic communities.

The software industry study team leader is Prof. William Miller, Director of SCIP. Our research assistants are Marisa Quinn and Jim O'Brien.