Software Technology and Industry of Korea: Widening Horizon and Emerging Presence

Yong Rae Kwon
Department of Computer Science
Korea Advanced Institute of Science and Technology

Abstract *Software organizations in Korea have accumulated confidence through their development experience over the past 20 years. However, software development is of limited types and has not been faced with stringent productivity and quality requirements until recently. Software houses undergoing a difficult period due to a narrow software market can ill afford to invest in research. Craftsmanship, not engineering, is still a predominant mode of development. Researchers in software as well as software engineering are having difficulty defining projects with concrete goals and well-organized teams as demanded by funding sources. Upcoming large scale software projects initiated by the government are expected to create a lot of research opportunity and thereby drive the domestic software technology to a higher level.*

Software Industry Software developers in our country have managed to meet most of their data processing needs relying on their own capability. They have piled up confidence in development with hard work and previous experience rather than with modern software techniques. Most software development was conducted in-house and many of these in-house organizations have evolved into software houses in the past few years. Some of the software houses participated in development of software systems for the 1988 Seoul Olympic Games and exhibited their products to foreign atheletes and visitors. They received international acclaim with their products and their success enabled them to firmly mark their presence in the country as well. They recently formed a software industry association and are lobbying for various government supports and are discussing joint research and development efforts.

Legislation for protecting development programs was enacted by the government in 1987 and over a thousand programs have been registered since then. Those measures will undoubtably encourage the software houses to vigorous development efforts. However, the most serious problem facing them is the domestic software market. Word processors supporting the Korean language are some of only a few marketable commercial products at the moment. Without prospects for significant sales, software houses can hardly afford investment to refine their development techniques. Measures to promote software development and to subsidize the software houses are in preparation by the government.

Despite a slow start, the domestic software market is projected to grow on the average 33% per year to 9.3 billion U.S. dollars in 1991. Several big government projects are expected to create sizable jobs for the software houses. The National Administration Information System (NAIS) and planned military projects are of unprecedented scale by our standards. Aiming to computerize government administration and public services, the Data Communication Corporation manages its projects with major software houses as subcontractors. Needs for customized programs for small computer systems are growing rapidly and have opened bright prospects for small, but adventurous software firms. Compulsory computer education begun in elementary and junior high schools in 1989 will also help to vitalize the software market.

Lacking an urgent need for productivity enhancement and stringent quality requirements, domestic software efforts have not encountered discouraging failures. Interest in modern development techniques is steadily rising, but is not vindicated by actual deployment of them. Management procedures such as cost estimation, quality assurance, and configuration management have begun to be practiced.

Software Technology Domains of software development have been largely in the area of information systems. Full-fledged system software development has been initiated only recently. An operating system and a relational database system for a multi-processor supermini system for the NAIS project are being developed by the Electronics and Telecommunications Research Institute (ETRI). A consortium of software houses is jointly developing an operating system for personal computers. Compatibility, standardization, support for Korean language processing are major concerns. Traditional engineering fields are satisfied with just executing imported software packages at the moment. However, soaring wages and domestic currency value are forcing many companies to speed up their schedules for automating their facilities.

ETRI successfully completed TDX-1, a telephone exchange software system, in 1987. It was an outcome of

several years of strenuous development efforts. In terms of the size and complexity of the projects, this TDX project is the most significant accomplishment in the history of domestic software technology. Software engineering approaches were not employed much in this project, but the need for them was keenly perceived. As a result, a programming environment, specialized for the CHILL language, has been developed since the completion of the TDX-1 project and is planned to be extensively used in the development of the next-generation TDX.

Expectation for software engineering techniques was high as a means of enhancing software productivity when the Ministry of Science and Technology selected the information industry as a strategic export industry a few years ago. Unfortunately, the results of ensuing software engineering research projects have not received favorable responses from the software industry partly because the industry was not ready to accomodate new techniques. A research project on software environments will always be a possibility considering the increasing concern for software productivity. In fact, the Systems Engineering Research Institute (SERI) has launched the Software Usability and Productivity Enhancement Research project in 1988. It is an ambitious three-phase project aimed at constructing a software factory by 2001. However, it will take some time for SERI to refine its goals and lay out a detailed phase plan so that the software industry and universities could define contributing roles. SERI itself is undergoing a major reorganization.

Software researchers have managed to claim only a small portion of the research funds available for all information technologies. They have problems defining well-focused research goals and convincing goal-oriented government officials into sustained support. Many people still do not understand that software research typically takes a longer time to arrive at a goal and the goal itself tends to be abstract. Several new projects have been supported in the past few years. One example is machine translation between Korean and English or Japanese. Immediate needs and potential commercial value have induced many academic and industry research teams to accelerate their efforts. Building computerized dictionaries and sharing results from related research efforts will definitely require good project management. Research on Korean character recognition has been continuously pursued by several university and industry teams as data entry needs soar. Commerical products are expected to be on the market soon.

Future Issues Undoubtedly, giant strides have been made since bona fide software research was initiated and the software industry appeared in Korea several years ago. However, certain obstacles must be overcome before software technology and the software industry mature. Some issues are the following:

o Skilled software engineers are scarce in Korea and the few seasoned experts prefer universities or research institutes to industries. Students graduating from the universities usually find unfamiliar work settings in industry. The software industry complains that college education in computer science is not tailored to fulfill industrial needs and therefore their on-the-job-training takes too long. A graduate school specialized in retraining and conversion programs in information technology has been proposed by industry. A common denominator of technological environments should be defined so that utilization of the limited manpower could be maximized. A remedy to improve this situation is to provide universities with better computing environments. Without unlimited access to computers, students are bound to have difficulty not only learning programming skills, but also understanding software issues.

o The man-machine interface of English-oriented computer systems is awkward for many Korean users. Programming languages and operating systems need to be tuned by taking the characteristics of the Korean language into account. Rearch on recognition of non-English voice or characters as keyboardless input mechanisms needs to be continuously pursued. Naturally, such problems cannot be solved simply by adopting techniques developed in English-speaking countries. Imported software per se cannot be introduced to users without providing a Korean language interface.