OCCASION

This publication has been made available to the public on the occasion of the 50th anniversary of the United Nations Industrial Development Organisation.

DISCLAIMER

FAIR USE POLICY

CONTACT

Please contact publications@unido.org for further information concerning UNIDO publications.

For more information about UNIDO, please visit us at www.unido.org

UNITED NATIONS INDUSTRIAL DEVELOPMENT ORGANIZATION
Vienna International Centre, P.O. Box 300, 1400 Vienna, Austria
Tel: (+43-1) 26026-0 · www.unido.org · unido@unido.org

17693

Distr.
LIMITED

ID/WG.478/1(SPEC.)
30 August 1988

ORIGINAL: ENGLISH

United Nations Industrial Development Organization

Global Preparatory Meeting for the
First Consultation on the Electronics Industry

Grenoble, France, 28 November - 2 December 1988

SOFTWARE INDUSTRY: DEVELOPMENT APPROACH*

Prepared by

Seongjae Yu**
and
Yangyul Kim ***

47/5

* The views expressed in this document are those of the authors and do not necessarily reflect the views of the Secretariat of UNIDO. This document has not been edited.

** Professor and Dean of Business School, Chung-Ang University, Seoul, Korea

*** Professor of Economics, Sung-Kyun-Kwan University, Seoul, Korea

V.88-28127

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

INTRODUCTION

1. Many industrial observers agree on that the world is now in the midst of the fourth industrial revolution which is equal to, if not greater, the previous ones. The on-going revolution is a result of the development of Information Technology (IT). Its impact is enormous on the industrial activities, and human life style as well. The IT industry is huge and complex. It is growing and changing rapidly. Computer software is a vital component of the IT industry. All of the hardware components of IT are dependent on software to accomplish their functions.

2. Transition to the emerging new area is generally recognized to be difficult for many developed countries. It is even more so for the developing countries who have not yet prepared themselves to jump into such an industrial stage. However, the computer software industry is very appealing to the developing countries, not only because of its large growing market all over the world, but also because of its far-reaching impacts on the future economy. It also requires a minimum of natural resources, and yet significantly affects productivity of other industries. For these reasons, it has attracted the attention of policy makers in developing countries.

3. Most applications based on information technology are structured around and operate via their software components. These consist of a combination of data and instructions carried in the form of binary electronic signals that are interpretable, and executable by the system. As systems become more sophisticated, diversified and widespread, the software tends to become more complex, more diversified, and to be produced in a constantly increasing number and variety of areas in the industrial structure. As a result, software is expected to be of increasing importance in the technical systems of a nation and its production is becoming an industrial activity within its own economic structure.

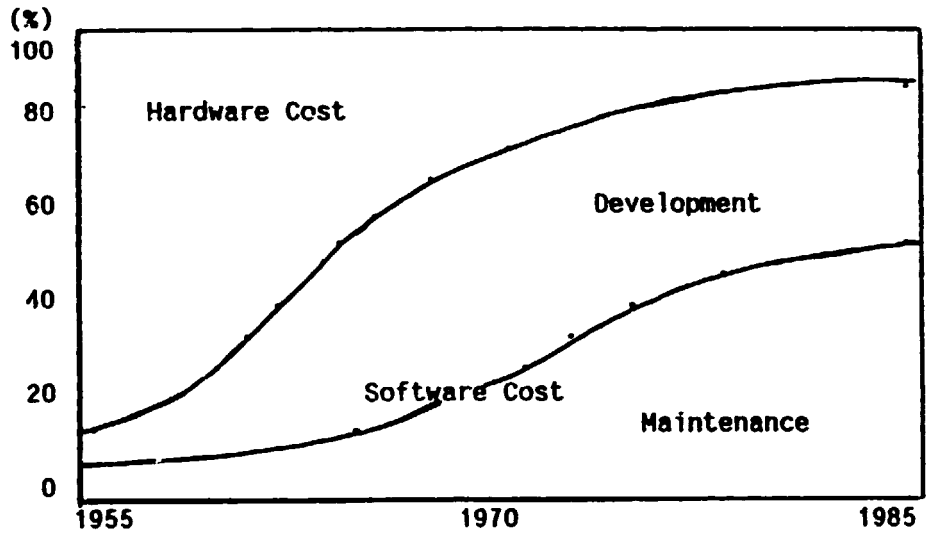
4. With the rapid technological progresses in hardware components of the computer system, the software side is clearly, today, the main constraint on its potential applications. It also impedes the system costs down and the diffusion throughout our economies and societies. This software bottleneck is illustrated by the increasing trend of the software expenditures relative to the hardware. It is estimated that software now averages over 80 per cent of the total life cycle costs in today's computerized systems.[1] (see Figure 1) This is due in part to the rapid advances in

---

[1] ADB, Technical Assistance Study of Selected Industries, Vol. 4 (Computer Software), 1987.

hardware technology where the price of a given item is rapidly decreasing and the performance for a given expenditure is rapidly increasing. More powerful and less expensive hardware makes larger and more sophisticated programs possible. Meanwhile, the labor cost for programming works continues to rise, reflecting the market forces of demand exceeding supply, and the productivity in software development which is not significantly improved.

Figure 1. Hardware/Software Cost Ratio (US)



Source: KIIA/Sperry, 1985; and many other references in OECD(1985).

5. The objective of this Study is to provides governments and business communities of the developing countries with basic information on the future global market and technological trends in the computer software industry, and an objective assessment of the potential for the development of software industry in the developing countries, as well as policy options and strategies for the development of the industry.

6. In line with the objective specified in the above, the Study shall encompass an analysis and appraisal of the global software industry in terms of the market and technology and their impact on industrial development in developing countries. Specific details shall be provided for the Republic of Korea as a case study. In the first chapter, characteristics and structure of the software industry shall be discussed as a guide to the industry. The second and third chapter deal with the global overview of the market and technological trends in the software industry. In the following chapters, the author reviews the promotional policies in the advanced countries and Korea as a case, before discussing implications of various policy measures to the developing countries.

Chapter 1

SOFTWARE INDUSTRY: DEFINITION AND CHARACTERISTICS

7. Computer software and software packages are referred to a collection of programs which work together to accomplish one or more tasks (hereafter, simply software will be used). In other words, computer software is the collection of machine readable instructions which control the actions of electronic computers. At its most basic level, software refers the logic which makes a machine perform work. According to the World Intellectual Property Organization (WIPO), "computer software" is defined to include all of the computer programs, program descriptions, and supporting materials. A computer program is a set of instructions by which a machine having information-processing capabilities performs a particular task. Program descriptions mean a complete procedural presentation in verbal or other forms, describing a set of instructions of a corresponding computer program. Supporting materials mean any materials other than the above two, created for aiding the understanding or application of a computer program.[2]

8. Computers are programmable general purpose machines. They are built to be programmed to do something specific. Computer software is useless without hardware equipment and hardware equipment can do nothing without software. Therefore, understanding the computer software industry requires an overall picture of the information industry.

9. Figure 2 shows the status of the computer software industry within information industry. Information industry is huge, and rapidly growing. The industry is expected to lead the other industries in near future. Subsectors of the industry are not totally independent each other. Rather, they are closely interrelated. The software industry plays the most important role among the subsectors.

A. Structure of Software Industry

10. It is safe to say that the software industry is composed of three: software suppliers; human and physical resources relating to the software production; and software products/market. Figure 3 depicts this system.

---

[2] OECD, The Emerging Industry: Software, 1985.

Figure 2. Structure of Information Industry

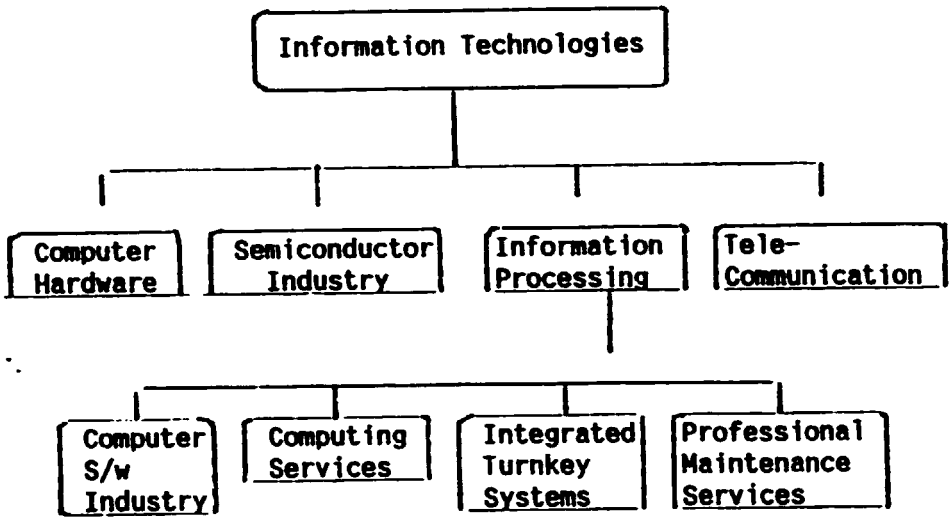
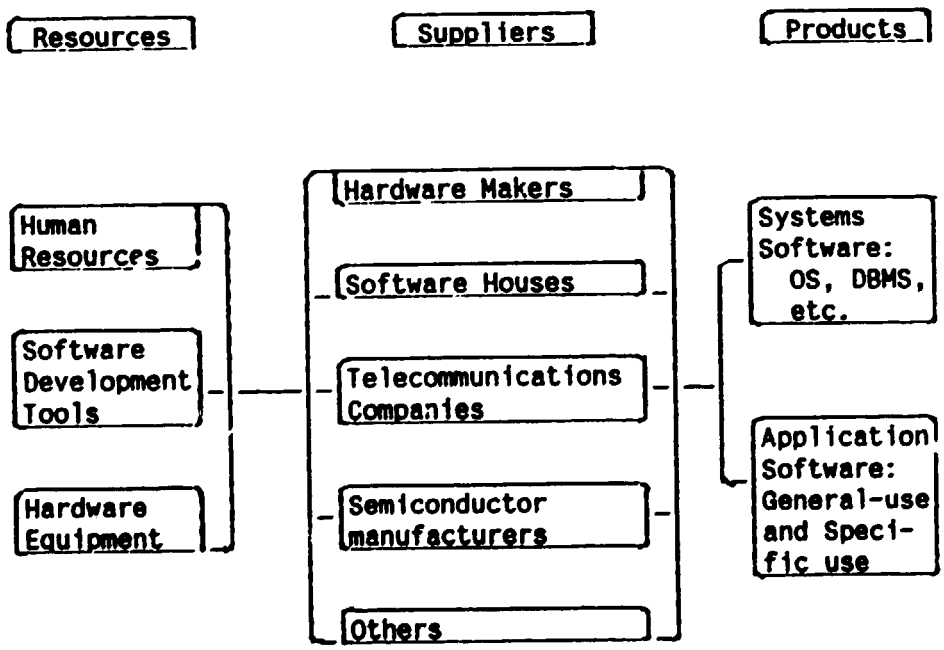


Figure 3. Structure of Software Industry



Source: KIET/US Dept. of Commerce, A Competitive Assessment of the US Software Industry, 1984.

(1) Suppliers

11. Computer software is developed and supported by hardware manufacturers who build or sell hardware equipment, user companies who do not intend to sell software but want to use it themselves, and specialized software houses who specialize in developing marketable software.

12. Naturally, most of software products in the markets are supplied by computer hardware manufacturers and independent software houses. Large hardware manufacturers supply computer software to improve the usability of their hardware systems and as a part of customer services. At present, they take a large portion of software market for main/mini computers.

13. There are numerous computer software development and marketing companies in the world. Software companies are small in size, compared to the other industries. Though there are some multi-national companies with tens of thousands of employees whose primary product is computer software, it is common to observe companies with one individual working part time on evenings and weekends. As the microcomputers are used widely and the software demand for them increases, independent software houses have been popular, especially in US, counting more that 7,000.

14. Supply by independent software houses is growing continuously relative to that by hardware manufacturers. Meanwhile, as the integration of industries progresses, a new class of software suppliers is emerging: a telecommunication company American Telephone and Telegraph (AT&T) supplying UNIX operating system; semiconductor companies developing microprocessors; large publishing companies providing educational software.

(2) Resources

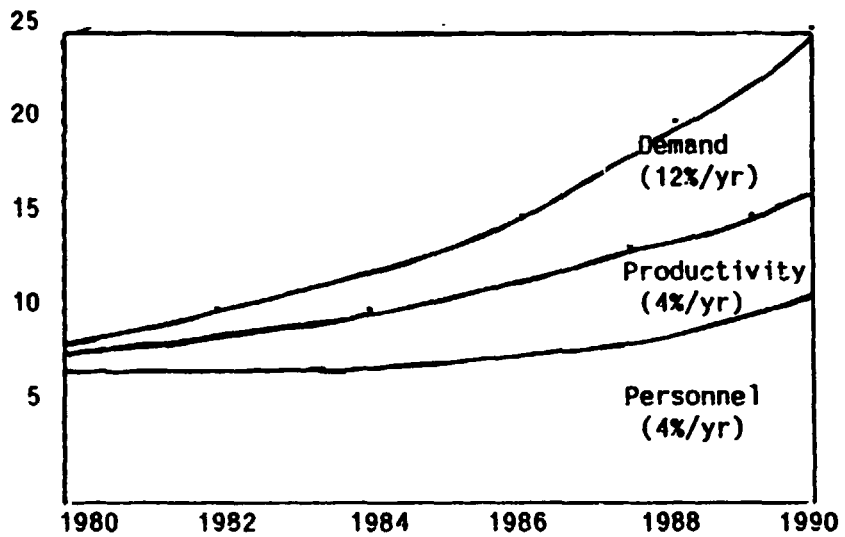
15. Software technical manpower such as systems analysts and programmers is the most critical resources for software development. Systems analysts, also known as software designers, generally analyze a problem that might be solved by a computerized system, then design a computer program to deliver that solution. Based on the design of the systems analysts, a programmer uses his knowledge of programming languages to write a series of instructions in a computer programming language.

16. Computer programming is a highly labor-intensive undertaking of which productivity grows rather slowly despite the huge leaps in hardware performance. Due to the rapid increase of software demand by widespread computers, the gap between supply and demand of

software manpower is widening. This may be one of explanations of so-called 'software bottleneck.' (see Figure 4)

17. Most commercial software products are incessantly debugged and improved. Fixing bugs is referred to as maintenance. The act of improving software is called enhancement. Both require significant man-hours.

Figure 4. Demand-Supply Forecast of Software Manpower in the United States



Note: The vertical axis indicates million personnel equivalent 1980.
Source: OECD/DOD, 1983.

18. In relation to the shortage of software manpower, a variety of tools for increasing software productivity are being developed. High-level languages, software development systems, application generators are examples of such tools which enable automation in software development. So far, automation has been achieved in the coding which covers about 10% of total process. Another resource consisting of software industry is hardware equipment. A specialized system is required in software development and for testing and debugging developed software.

B. Classification of Software

19. Though there are no definite criteria of categorizing software products, the classification by use is most widely applied. Based

on this criterion, software products are largely divided into two classes: systems software and applications software. While systems software aids programmers through controlling internal hardware systems, applications software is used to undertake specific jobs such as credit management, computer-aided instruction, software development, etc. Applications software is further divided into specific-use programs and general-purpose software.

20. Another dimension of classifying software is the product type, which divides software into packaged software, customs software, and integrated system. Packaged software is usually a standardized product, which is used without any alteration. Customs software is developed to meet particular needs of a user. Integrated system is usually supplied by system developers who incorporate their software into the hardware equipment externally owned. This type of business is a newly emerging area. Software is also grouped in terms of the size of computers on which it runs.

21. Because of the short history of software and the very rapidly changing technological environment, no single criterion for categorizing software is dominating yet. Industrial economists adopt a criterion at their convenience and according to the analytical purposes. In this Report, the classification by use will be mostly applied.

C. Software Production Process

22. The software production process is a succession of different stages requiring differentiated skills, methods and techniques. The stages are broadly grouped into system definition; programming; testing and debugging; maintenance and enhancement. Among them, the last is an integral part, continuously adapting software to new needs and situations. In the case of packaged software, the effort is revealed through successive releases of the same package, each of which upgrades and enhances earlier versions.

(1) System Definition

23. System analysts first draw an overall picture and logical flow of the problem they are facing. In this stage, the detailed specification for the software is defined including inputs, logical structures and outputs. Based on the specification, flow-charts constituting the skeleton of the software is prepared.

24. This is the most critical phase in the software production process. But, little technical progress has so far been achieved in this stage, partly because of the intellectual and organizational complexity of the phase. At present, many researchers delve into the application of artificial intelligence techniques, for the purpose of improving productivity in this phase.

(2) Programming

25. In this stage, programmers use their knowledge on programming languages to translate the procedure as defined by the analyst into a set of machine readable instructions. The language used by the programmer to communicate with the computer may range from a low-level assembly language to one of the high-level languages in which a single instruction calls for a whole series of instructions or sub-programs. Despite the considerable advances in productivity by the recent development of high-level languages, the programming stage is still very time-consuming.

26. The most commonly used languages today were developed in the late 1950s and some in the 1960s. The most important are Cobol, Fortran, APL, PL/1, Basic, and recently Pascal and C language, of which Cobol is the most widely used language. Recent advances in programming languages are noticeably improving the productivity of the software development process. These advances are in the direction of the structured programming and fourth generation programming languages.

(3) Testing and Debugging

27. The third stage is to test the developed software, in order to identify and correct bugs made in the previous stages. This generally requires considerable time and resources. Commonly confronted with the almost impossible task of going through tens of thousands of instructions with a fine toothcomb to detect the source of errors, software designers are trying to alleviate the operation by improving the reliability and quality of work during the preceding phases. This stage in particular requires great skills and experience.

(4) Maintenance

28. The National Bureau of Standards (NBS) reported in a study that only 33 per cent of expenditure in software over its whole lifecycle was for the first three stages, as against expenditures for maintenance amounting 67 per cent. This stage is regarded as one of the main causes of the software bottleneck, since it limits the resources available for the development of new, more efficient programs better geared to user needs. An estimated 75 per cent of computing resources for 1980 in the United States were taken up by maintenance activities, leaving only 25 per cent for development activities.[3]

29. Maintenance activities are largely attributable not to shortcomings

---

[3] OECD, Software: An Emerging Industry, 1985

in the finished software, but to alterations in adapting to its external environment, i.e, hardware changes, systems software changes, new user requirements etc. However, many experts agree on that if more effort was made in the early stages of the development process, less resources would have to be spent on the maintenance stage.

30. Computer software is not consumed. It may be used indefinitely with adequate maintenance activities. Most software products are not sold, but the developer sells a licence to use them. Unlike most products which wear out and become less valuable as they are used, computer software products gain value with age and use. This may be explained by increasing familiarity with the package as it is aging, and stronger user bases as it is disseminated widely.

D. Industrial Characteristics

(1) Technologically Formative Industry

31. Software production at present largely depends on programmers' skills, a sort of arts. It does not require a real technology. Software design is based mainly on the intuitive and experimental skills of specialists. They make but little use of theoretical tools to improve productivity of their work. Among the software development stages discussed previously, it is mainly the programming that has so far achieved substantial technical progress, through the development of many high-level languages. Virtually no progress has been appeared in the development of software engineering aids and methods to improve software design, quality and maintainability.

32. This fact limits the performance that can be expected from current software development, particularly for complex applications. Furthermore, since software design is very largely an intellectual task which has little room for automation, software production is at present extremely time-consuming, labor-intensive.

33. Manpower is the most important factor for development of software. However, unskilled labor is easily trained to take low level programming works. About three months of training is enough to get involved in software development. More advanced skills are achieved on the job training. This makes the software industry more competitive in certain areas.

(2) Uncertain Economic Status of Software

34. The great majority of software is still produced in house development teams that use it or incorporate it in their products or services. Thus, it is not subject to market competition, and is not fairly valued. In addition, software can be reproduced very easily

and very cheaply. This makes a difficulty that software development activities are not yet perceived by most users as a production activity. The invisibility of software further makes it difficult to value.

35. These factors may hamper the formation of the market for invisible technical endeavors, in which they are disseminated efficiently. In particular, it may discourage user organizations having their software needs met externally, i.e. using the expertise and products of independent suppliers. With the efficient market for software, the independent suppliers may take advantage of the considerable economies of scale typical in software production, and competition among them increases the quality of the services and products and forces prices down. This will make the market expand.

36. There are uncertainties with regard to the way in which intellectual property law, accountancy law, and tax law apply to software. It is not yet clear how software products should be protected by copyrights and patents. However, there is an increasing trends that software is protected by copyrights. Whether the software development costs can be regarded as R&D investment expenditures is also controversial in accounting and tax purposes.

(3) Unstable Software Products/Factors Market

37. Markets for software factors such as skilled labor and software engineering tools, and markets for software products such as professional services and software products are at present immature. This again causes the market to function poorly as channels for technology transfer and rational economic decision-making.

38. The market for skilled labor shows significant imbalance between supply and demand. (see Figure 4) The rapid changes in users' technical requirements, in hardware and software environments, call in fact for rapid updating in the skills of specialists. This requires training of more up-to-dated skills. However, the quantitative pressures on the labor markets make this qualitative adjustment more difficult. The variety and rapidity of technical progress give rise to difficulties in defining and certifying professional skills, which also hamper the establishment of the stable labor markets. On the other hand, demand for software engineering tools is still very low, mainly due to the delay in adjusting the skills of potential users.

39. The instability of the market for software packages results mainly from the rapid technical advances on the supply side and substantial uncertainties on the demand side. These uncertainties comes from the proliferation of alternative products offered and by the more and more interdependent and less and less reversible nature of software investment within organizations. Some excess of supply over demand is also observed in certain software package markets.

E. Some Positive Aspects of Software Industry

(1) Informatization of Industrial Structure

40. It is well noted that the industrial maturity will lead to information society. A visible driving force for the informatization of industrial structure is the technological advances in semi-conductors and computer technologies. Software works as a catalyst for informatization by systematically combining various levels of industry sectors. The progress of informatization is observed in the increasing weight of information sectors in an economy. In Japan, the share of information sectors increased from 29.5 per cent in 1960 and 36.9 per cent in 1970 to 43.5 per cent in 1980, showing the increasing trend of information industry. This is a general trend observed all over the world.

41. Along with the informatizing trend of economies, development of software industry is indispensible for any economy to be competitive in the international markets. Because software is an indispensible factor for information equipment which is inducing information industry. The equipment can do nothing without software.

(2) Growing Market and High Value-Added

42. Software industry is most rapidly growing among information technology industries. The US software market in 1986 was $14.8 billion, but it is expected to grow to $52 billion in 1991 with an annual growth rate of 22 per cent. (see Table 13) The industry is expected to grow in other economies too with an even higher rate. That means software industry may well deserve a due attention for industrial promotion.

43. Furthermore, it is generally agreed on that the value-added potential of software industry is very high. Though there is no verified hard data on the statistic par' / due to the wide variation of the software sales price, the high va.ue-add in software industry is very likely from the fact that most of software development costs are for labor, and the portion of materials and equipment usages is rather small.

(3) Pervasiveness to Other Industries

44. With the informatization of industrial structure, production processes of goods or services become increasingly automated in all areas. Technological advances in microelectronics has enabled the introduction of such automation systems, which in return increased productivity significantly. Industrial automation or networking of manufacturing systems is not possible without proper supports of

software. Software contributes not only to the productivity increase of the information industry, but also to that of various manufacturing or service industries.

45. So far, development of software is mostly hardware-induced, but the situation is changing. That is, though up till now, software development has followed a new hardware architecture development, with the sophistication of user needs, software will require a specific hardware environment, leading a new hardware technology development. This revolving process will lead to an overall revolution of information technology.

(4) Big Potential for Job Creation

46. The software industry has a big job creation potential in which most governments are interested. In 1982, the US computer services industry was employing some 450,000, the French industry 34,800, the UK industry 33,900 and the German industry 23,600. This means an average annual turnover per employee of $60,000 in the United States, $49,000 in France, $44,000 in the United Kingdom and $52,000 in Germany.[4] This relatively low labor productivity can be explained partly by the low capital intensity of the sector, which implies more jobs created for a given amount of investment. Development of software requires manpower with various degree of skills. The current backlog of software, observed in advanced countries, is mainly due to the manpower shortage.

---

[4] OECD, 1985.

Chapter 2

SOFTWARE MARKET: TRENDS AND PROSPECTS

47. Due to the wide range of software products and the diverse user organizations such as manufacturing industries, government authorities, financial institutes, etc., it is very difficult to measure the size of computer software industry. It is fairly often to see industrial data to be published only on a particular segment of the computer software industry, i.e., markets for mainframe computer systems software, markets for database management systems (DBMS) for minicomputers, PC word processing software markets. The software market forecasts commonly have substantial margin for error in them. Adding the projections for all segments of the marketplace together for industry-wide forecasts is difficult and risky.

48. A difficulty in getting accurate figures on computer software stems from the fact that at least half of all computer software is developed within an organization for internal uses, and is not traded in the market. Most published data are based on sales or shipments by software developers. The amount of software demand out of the market seems to be great. Though some estimate around a half of the software demand is satisfied by in-house development, no concrete estimation is available. This may explain the substantial margin for error in forecasts. Another large portion of software is not sold separately from the hardware products incorporating it, making the economic value of the invisible software nearly immeasurable.

49. Another difficulty is from its uncertain economic status. Software products can be reproduced at very low costs. Due to this characteristic, software is regarded as a free good in many cases, and is circulating free of charge among users. The market obviously exists. But, we can hardly measure its size in economic terms.

50. Despite the difficulties, some research organizations challenged to measure the software market size in amount. But, the estimates of worldwide software market in 1986, range from $45 billion to $100 billion in purchases and development expenses.[5] Forecasts of the global software market in the year 2000 range from $80 to $200

---

[5] ADB/Business Week, March 2, 1987 and Computer Design, Jan 1, 1987.

billion.[6] The wide range of estimates and projections reflects the difficulties in measuring software market, and uncertainty of market growth and technological advances.

51. In the following, a few distinctive features observed in the demand and supply structure of software will be discussed. Before closing this chapter, a brief review is given on the factors changing the software market.

A. Major Trends of Software Market

52. As mentioned previously, software may be categorized in terms of the size of hardware on which it runs. Presently, many software houses partition the market in this way, and develop marketing strategies accordingly. A noticeable characteristic in the current software market is the explosively growing market for micro-computer software. Along with the increase of sales of packaged software, the portion of PC software is projected to increase from 15 per cent in 1982 to over 25 per cent in 1991. This increase is mainly due to the widespread use and the increasing capacity of micro-computers by the development in hardware technologies. This issue will be covered in detail in the following section.

53. The second characteristic observed in the market is the variations in the regional market structure. While the United States covers over 70 per cent of the global market for packaged software, software companies in France and Japan, and other major countries, are concentrating in customs software. (see Table 1) In 1982, while in US, packaged software consisted of 56 per cent of the total sales, in France and Japan, customs software occupied 70 per cent and 95 per cent, respectively. In the United Kingdom, integrated systems sales amounted 54 per cent of the total.

54. Along with the strong demand for general-use packages such as spread sheet software, there is a sharp increase in industrial demand for packages designed for special applications. This resulted from the specialization of within and inter-firm developmental activities, which in turn, accelerate the specialization of software houses.

55. Owing to the growing packaged software market and the mass dissemination of applications software, independent software vendors increasingly lead the market. In 1981, independent software vendors covered approximately 43 per cent of the total sales in the United States, but the share increased to 49 per cent in 1983 with the annual growth rate of 48 per cent. (see Table 9 and 10) Table 2 also shows the share of independent vendors in Europe. Independent vendors take 65 per cent of customs software, and 47 per cent of the total European software market.

---

[6] ADB, Technical Assistance Study of Selected Industries, Vol. 4 (Computer Software), 1987.

Table 1. Major Areas of Software Development in Advanced Countries (1982)

$ billion

	Sales revenue	Major area	Global market share (%)
USA	10.3	Packaged s/w (56)	70
France	1.3	Customs s/w (70)	5 - 7
Japan	1.2	Customs s/w (95)	5 - 7
UK	0.7	Integrated s/w (54)	2 - 3

Notes: Numbers in parentheses are per cent.
Sources: EIAK/Information Industry Yearbook (1986), Korea.

Table 2. Western Europe Software Market Structure (1983)

$ million

	Packaged software	Customs software	Total
Hardware manufacturers	1,885	549	2,434
System houses	497	682	1,179
Independent vendors	983	2,261	3,244
Total	3,365	3,492	6,857

Source: OECD/ECSA, 1984.

56. The concentration in packaged software by the American software houses which are leading the software industry technologically, is explained by the low productivity of the programmers on one hand, and low reproduction cost and relatively high profitability of packaged software on the other.

(1) Wider Acceptance of Packaged Software

57. Data processing expenditures are growing very rapidly all over the world, with different consumption structures from country to country. Rather than differences in the factor costs, the differences are explained by the different dem. l structure. Some

information is available about how demand breaks down between customs software and software packages, through the software supply structure. Table 2 suggests that the packaged software occupies almost a half of the total European market in 1983, but its share in Germany is expected to be 70 per cent in 1990. (see Table 3)

Table 3. Software Demand in Germany (1985 - 1990)

$ million

	1985	1990	Growth rate
Packaged software	1,157	5,198	
Hardware manufacturers	625	2,794	35 %
System houses	165	545	27 %
Independent vendors	367	1,859	38 %
Customs software	707	2,260	
Hardware manufacturers	128	361	23 %
System houses	170	460	22 %
Independent vendors	409	1,439	29 %
Total	1,864	7,458	

Source: ADB/Computerworld, CW Communication, 1987.

58. A gradual increase of the packaged software market is a visible trend, as more personal computers are used in more sites. It enables several data processing requirements to be satisfied at low cost. Offering the user an appreciably wider procurement horizon, software packages also enable him to benefit from advances in the state of the art. In the effort to make optimum decision in utilizing data processing resources, more organizations go out to the market for their software needs.

(2) Growing Demand for PC Software

59. Today, personal computers are everywhere. They are now used in nearly all organizations in various fields. As of the mid 1987, the IBM-PC and its clones number over 10 million units in the USA and there should be at least half again that number in the rest of the world. The Apple Macintosh, which has no clones, has just over one million units in service. Although not many users have the personal computer in their full control, and master more than one program, the systems have so much capability by which a very beginner may be rewarded with worthwhile outputs for a short period of practices. In this environment, the opportunities abound for personal computer software houses.

60. The software market for microcomputers is expected to grow with an average rate of 30 per cent until it reaches $45 billion in 1990. (see Table 4) The United States takes a great part, 65 per cent, of the global market. The market in economies outside US is more rapidly growing.

(3) Widening User Base

61. So far, computers have been used mostly in large business and government organizations. Small companies cannot afford that expensive computing facilities. Benefited from the recent development of microcomputers, software packages and small turnkey business systems, however, a great number of small firms are increasingly reacting and investing in computing equipment. The market for small business systems is the most rapidly expanding. (see Table 5)

Table 4. Packaged PC Software Market (1984-1990)

$ billion

Year	Total	US	Outside US
1984	11.1	8.4	2.7
1985	13.2	9.7	3.5
1986	16.2	11.6	4.6
1987	20.5	14.3	6.2
1988	26.3	17.9	8.4
1989	34.2	22.7	11.5
1990	45.0	29.2	15.8

Source: ADB, 1987.

Table 5. Computer Hardware Sales Forecast in the United States

$ billion

	1987	1988	1989	1990
Mainframes	13.2	14.2	15.3	15.9
Mini-computers	16.8	18.5	20.7	23.2
Multi-user micros	4.5	5.7	7.2	9.2
Desktop micros	14.5	16.7	19.7	23.0
Portable/laptop micros	4.5	6.4	7.9	9.5

Source: ADB, 1987

62. The smaller firms may introduce relatively easily computing facilities with the more efficient configurations. They do not have the problems facing large firms such as obsolescent computing stocks in both software and hardware. But, they must also heavily rely upon outside service companies for software products and services, because they cannot build up the full range of resources necessary to operate computer systems. Software houses played a significant role in the development and dissemination of software for this group.

(4) Emerging Software Support Markets

63. The software support market is closely associated with the maintenance phase in software development. User organizations contract with software publishers or independent parties for technical support of software packages. The technical support consultants supposedly help the customer solve whatever difficulty he has come up. If the publisher releases an updated to the program, a customer with a software support contract is generally entitled to a no-extra-cost copy during the time that the support contract is in force.

64. As computers are used more widely, and software becomes more sophisticated, demand for this type of services is rapidly growing, especially in systems software. The market is expected to amount $6.8 billion in 1991. (see Table 6)

(5) Market for Artificial Intelligence

65. Currently, expert systems are the typical applications of artificial intelligence (AI). Several software packages of expert systems are already on the market, and the market for expert systems is expected to grow with predicted annual growth rate of 29 per cent per anum through 1990. (see Table 7)

Table 6. US Software Support Market Forecast by Product Type

$ billion

	1986	1991
Applications software	0.8 (35)	2.8 (41)
DBMS	0.4 (17)	0.9 (13)
Systems software	1.1 (48)	3.1 (46)
Total	2.3(100)	6.8(100)

Note: Numbers in ( ) are percentage.
Source: ADB/PC Week, Feb. 17,1987.

Table 7. Forecast of US Market for Expert Systems

$ million

Year	amount
1986	500
1987	646
1988	835
1989	1078
1990	1800

Sources: MIS Week, Feb. 9, 1987.

66. Computer programming opportunities in AI include writing general-purpose AI based software, writing specific applications in an AI programming language, and knowledge engineering such as translating data and rules from human experts into rules usable by AI expert systems. This market is expected to grow rapidly in various professional areas.

(6) Ada Market

67. Ada is a computer programming language conceived by the USA Department of Defense (DOD). DOD holds strict control of all copyrights relating to Ada. DOD, therefore, has full control of implementations of the language. This is important because each implementation of all other languages (Cobol, Pascal, Assembly etc.) have idiosyncracies that prevent them from being 100 per cent compatible with other implementation of that same language.

68. The DOD has mandated that all new computer software written for any division of DOD shall be written in Ada. This implies that there will be a large market for software written in Ada for at least the rest of this century, although commercial use of Ada outside of DOD contracts is minimal at present. (see Table 8)

B. Software Supply Structure

69. In the preceding chapter, major constituents of software industry on the supply side have been briefly mentioned: the internal supply of software by users, and external supply via the market. The latter is further differentiated according to the nature and origin of the suppliers: hardware manufacturers; and specialized software companies. In this section, some noticeable trends in the software supply structure are reviewed by supply type.

Table 8. Ada Market Forecast

$ million

	1986	1991	1996
Commercial software	540	1565	3400
DOD for tools and training	150	1100	3600
DOD operational software	40	14000	32000
Total	730	16665	39000

Sources: MIS Week, Jan. 5, 1987

(1) In-House Development

70. At present, a great portion of software is supplied by in-house development in various organizations. According to Informatics, in 1976, only less than 15 per cent of all instructions executed on computer in the United States were software purchased from outside suppliers.[7] Until the mid 1980s, in house development was the main source of applications software for large business and government organizations.

71. As budgets for data processing in business and governments grew fairly regularly, data processing departments in the organizations developed in-house most of the applications software required. This was partly because the software market was not well established, and the outside specialists were not enough to meet their requirements.

72. Computer departments in various organizations now, however, increasingly focus on the maintenance and enhancement of software, and purchase more from outside vendors in the packaged form. A 1982 survey in the United States showed that only 19 per cent of software being run on microcomputers had been produced by data processing departments, as against 28 per cent by users and the remainder by outside suppliers. In case of systems software, since it was so complex and costly, most of it was designed and produced outside (mostly by the hardware manufacturers) in the form of software packages, and this trend still dominates.

[7] OECD, 1985.

(2) Supply by Hardware Distributors

73. In the early days of the information processing industry, it was inevitable for computer manufacturers to develop software in order to promote their hardware sales. Some of this was systems software controlling the machinery to run as designed, some was applications software for the specific requirements of customers installing computer systems.

74. In this period, the systems software was included in the price of the hardware and not invoiced separately, while applications software was usually the subject of a turnkey contract, and customs software as a part of their time-sharing services. Since 1969, when the United States Department of Justice required IBM to invoice its hardware and software separately, the hardware manufacturers began to confront competition from independent software suppliers.

75. The manufacturers' strength lies mainly in systems software, and in the general-purpose applications software such as accounting, management, payroll for main/mini computers. The manufacturers themselves develop most of systems packages for their hardware products, with their hardware development projects. But, for applications software they tend to make use of more products developed by software houses. They offer only software packages, and practically no customs software except for very large orders. The tables 9 and 10 below illustrate the share of hardware manufacturers in the software markets in the United States, and Table 2 for the European market.

76. Development of applications software requires a full understanding of the applications concerned. Since the hardware manufacturers are not particularly competitive in this sense, they approach and encourage the independent software houses to design software to run on their hardware and market it directly. Almost all the manufacturers have adopted this strategy recently. The availability of applications software plays an increasingly important role in the competitive strength of the hardware manufacturers, especially in microcomputers.

77. In spite of their leading position in the software market, software production is still a minor part in their business activities. But, many plan to increase this share significantly in the future. The most telling instance is probably that of IBM, which set up a large independent business unit in 1982 specializing in software sales, the Information and Programming Services (IPS). IBM aims to increase software sales up to one third of its total business by 1990s.

Table 9. Software Supply by Vendor in US (1981-1983)

$ million

	1981	1983
Hardware Manufacturers		
Systems software	2,025	3,660
Applications software	295	455
Independent vendors		
Systems software	820	1,854
Applications software	965	2,060

Sources: OECD/IDC/US Department of Commerce, 1984

Table 10. US Software Supply Structure (1983)

per cent

	Systems software	Applications software	Total
Hardware manufacturers	45.6	5.7	51.3
Independent software houses	23.1	25.6	48.7
Total	68.7	31.3	100.0

78. The manufacturers are developing strategies in various ways, adapting to the changing environment. They are trying to incorporate principles of distributed and integrated processing on their systems software. They hope both to make their hardware more competitive and to extend their share of the market, supplying software exclusively for it.

79. An area in which the manufacturers are interested is the development of firmware, software embedded in the hardware such as ROM (read-only memory). There is a good deal at stake. At present, it is mainly used in compilers, and to some extent in machines designed for a specific purpose. However, with advances in very large scale integration(VLSI) and in the design of specific circuits, the opportunities for applying firmware will extend in the future.

(3) Supply by Independent Software Houses

80. Information services are an emerging industry with a low degree of concentration. In 1986, the United States had nearly 7500 computer services firms with a total sales of $54.1 billion. (see Table 11). The United Kingdom has some 3000 firms in the industry, most with under 20 employees. In Japan, 2808 firms recorded 1915.9 billion Yens of sales revenue in 1986.[8]

Table 11. Revenue of US Information Services Industry

$ billion

	1978	1982	1986	1992
Processing services	5.58 (2089)	12.48 (2130)	21.32 (2110)	38.2
Software products	0.94 (752)	5.30 (1879)	14.78 (2705)	52.2
Professional maintenance services	1.23 (550)	5.33 (1348)	11.08 (1555)	33.9
Integrated turnkey systems	-	3.32 (1113)	6.85 (1162)	13.3
Total	7.75 (3391)	26.43 (6470)	54.08 (7532)	137.6

Note: Numbers in ( ) are the number of companies concentrating on the corresponding category.
Sources: OECD, 1985; IPA, Informatization White Paper, Computer Edge, Japan, 1988.

81. The number of software houses increased very rapidly since the early 1970s, in almost all countries. They are providing a wide range of services including: time-sharing services; consulting and engineering services; integrated turnkey systems; customs software and packaged software design and writing.

82. The software houses are still taking a minor share of total software production, compared to in-house production by users and software supplied by the hardware manufacturers. But they show the most vigorous activities technologically and economically. At present,

[8] IPA, Informatization White Paper, Computer Edge, Japan, 1988.

the major areas that specialized software houses are aiming at are the packaged PC software and customs software.

83. Software houses maintain a high level of skills at the disposal of various business and government organizations. Through their various services, software houses are helping them to exploit the new technology in data processing.

84. Services from specialized companies is becoming more important nowadays since it is impossible for in-house computer departments to evaluate and apply the state of the art by themselves. Software houses are also taking part in advancing software technology. Many important software innovations have been initiated by software houses, i.e., fourth generation data-base management systems, software engineering aids, microcomputer software.

C. Factors of Software Market Expansion

(1) Advances in the Computer Technology

85. Until the mid 1960s, the computer hardware market consisted only of mainframes. Computer users were limited to large business and government organizations. But, soon super-computers and minicomputers have appeared in the market. Furthermore, the introduction of micro-computers significantly changed the hardware market structure and computer user base. With the increasing variety of computer hardware, the software demand has also changed in nature and size. For example, the parallel processing, a technological characteristic of the super-computer architecture, induced software development for such a system. The development of highly sophisticated micro-processors and wide spread of micro-computers are also significant factors influencing the software market.

86. The most important impact of advances in computer technology is the wide dissemination of PC, especially for office. It has been resulted by smaller size, higher capacity, and lower prices of microcomputers. Furthermore, system integration of component equipment, increasing information network, increasing telecommunication function, information service, videotex, all these contribute to increasing software demand in the area from various organizations. The growth potential of PC software is the highest among all software.

(2) Progress in Office Automation

87. With the advances of information technology, development of highly automated management system is continued to be on the move in the 1980s. Office automation is referred to the utilization of office information equipment to increase the office productivity in

complicated decision-makings as well as routine repetitive jobs.

88. At the introductory stage of office automation, word processors, Faximille, or micro-computers come to be in use. Software demand at this stage is associated with using the equipment. The next stage is to integrate the information generated at all business units, connecting to the central computer system or data base. Thus, in this stage, mainframe systems software, database or interface software for distributed processing and variety of applications software are required. This is followed by the establishment of the network system which represents the most sophisticated office automation.

89. Through the network, every business unit shares information with various organizations, including banks, wholesalers, retailers, and others. This stage requires network operating systems and software such as LAN, as well as software mentioned previously. As the office automation spreads out more widely, software demand will continuously increase.

(3) Progress in Factory Automation

90. Factory automation is referred to a manufacturing system which handles efficiently ordering, design, manufacturing, and testing processes, applying latest information and mechatronics technology. Recent technological advances in programmable automation equipment and the computer integrated manufacturing (CIM), such as numerically controlled (NC) machining tools/centers, computer-aided design and manufacturing systems (CAD/CAM), and manufacturing resources planning (MRP), make the automation increasingly accessible for manufacturing systems.

91. Automation software is all programs governing the operation of programmable equipment in service to perform design, control, inspection and management functions in manufacturing processes. They are distinguished according to the types of equipment on which they run; NC machine tools; robots; measurement and testing equipment; CAD/CAE/CAM systems; MRP systems; process-control systems.

92. During the 1970s, with advances in information technologies, NC machine tools have been developed greatly and widely being used. A survey showed that the stock of NC machine tools in service in the United States required some 60 to 80 hours of programming per month per machine, and that the amount of time could increase markedly by the increasing sophistication of the machine concerned. Very approximately, it requires some 4000 hours of programming, costing far more than the machine's hardware itself. The software cost would even double in the next ten years, when obsolescence began to overtake them. This implies a vast amount of software demand in this area.

93. In case of industrial robots, it is not clear what proportion of hardware costs the software costs represent. According to an estimate, however, only system software accounts for over 50 per cent of the development costs of second generation robots (servo-robots controlling their own movement via sensors), excluding applications software or any data bases required for their use. Another consideration is that development of third generation robots (intelligent robots programming their own operations to achieve defined aims) will create even bigger demand, as they use sophisticated artificial intelligence techniques.

94. As another important area, CAD systems were developed to handle the vast amount of calculation involved in design of complicated systems. CAD systems consist of specially designed hardware components, but the most critical part is a combination of special software. That part of the software which is originally incorporated in the system is large-scale and complicated. It works by interacting with programs and data bases created by the user. So the main costs of developing and using a CAD unit are for software. Table 12 gives a brief picture of the CAD/CAE market, growing at the rate of over 50 per cent.

Table 12. Forecast of PC CAD/CAE Software Sales

$ billion

Year	Sales amount	Growth rate (%)
1986	1.0	-
1987	1.5	50
1988	2.4	60
1989	3.7	54
1990	5.6	51

Source: Computer Dealer, Daratech, Jan., 1987.

95. The CAD/CAE systems were developed rapidly during the second half of the 1970s through computers and memory devices constantly becoming more powerful and cheaper. The functions of CAD/CAE systems are grouped mainly in four: drawings and diagrams generations; Engineering calculations; Kinematic studies; Manufacturing instructions. They are now applied in many areas: automobiles, electronics, building, etc. Development of CIM and industrial robots has a great potential for creating new software demand in this area.

96. Owing to the changes in environmental factors contributing to the development of information industry, data processing expenditures in government and industrial organizations are expected to grow continuously. Considering the increasing capabilities and decreasing prices of hardware equipment, the share of the expenditures for software and services in the total data processing expenditures will increase rather rapidly. The following Table 13 provides a forecast of the demand for software and data processing services. The demand is expected to grow at an annual rate of over 20 per cent in most of the countries. A forecast for the US market is on Table 11.

Table 13. Data Processing Expenditures for Software and Services (1985 - 1990)

$ million

	1985	1990	Growth rate
Australia	1,040	2,310	19 %
France	3,493	9,111	21 %
Germany	2,769	9,034	27 %
Italy	1,591	4,277	22 %
Japan	3,950	15,239	31 %
Norway	458	1,085	19 %
Sweden	692	1,781	21 %
Switzerland	519	1,560	20 %
United Kingdom	2,600	7,000	22 %

Note : Here, software and services include packaged and customs software purchase and consultancy, training, processing services (batch and remote). About 57 per cent of the total was software in Europe in 1983.

Source: ADB/Computerworld, CW Communication, 1987.

Chapter 3

TECHNOLOGICAL DEVELOPMENT IN SOFTWARE INDUSTRY

A. Overview

97. The 1950s and the early 1960s, the introductory stage of electronic computers, were the age of machine and assembly languages. In this period, software business was just writing programs for solving problems with such languages. But, beginning in the late 1960s, the age of the third generation of hardware, with the wide utilization of high-level languages such as Fortran, Cobol, Algol, ordinary computer users could write programs and software industry started to move beyond simple coding. One area is the development of systems software which controls the computer system logically and physically. And the other area is the wide supply of applications software, both packaged and customs, which is used in solving practical problems.

98. In 1970s, with the development of hardware technology, software become very complex, and accordingly software development costs increased significantly. As a result, demand for standardized package systems increased, and general-purpose operating systems were begun to be developed. And applications software for solving a specific problems has become highly sophisticated from its early functions of numerical analysis, scientific calculation, and data processing, to management planning, automatic control of manufacturing process, ticket reservations, and space and atomic energy applications. That is, emphasis in software has been shifted from customs software to packaged software.

99. Since 1980s, due to the establishment of data base systems, and communication networks, together with the limitation of the large software systems for central processing, demand for distributed-processing software is constantly increasing. At present, software plays the central role in the various areas of information technology applications such as computers and word processors, factory automation, data-bank, telecommunications, banking, manufacturing, transportation control systems, and a variety of defense systems.

100. Development in software engineering technology such as software development tools, has started to make a wide stride. As a result of increasing emphasis on the efficient method of programming, testing, debugging, and program documentation in software production, a significant improvement in productivity and reliability is also observed. Nevertheless, system definition and

specification stage, which is the most critical part in software development, is still at its very early stage.

101. A latest advance in software is the appearance of firmware, which incorporates software into the hardware. This is usually done in ROM (read only memory). Firmware technology has so far been used in complex instruction sets for mainframes, in resident compilers, in language machines such as Ada machines and Lisp machines, and in some operating systems such as Unix for AT&T's 3B series. However, it is now applied in wide areas. A massive amount of applications and integrated software as well as systems software are embedded in a single chip, owing to the development of semiconductor IC technology. The development of super-chip is expected to affect software industry in diverse ways.

102. In the following, the trends in software technology will be discussed with respect to the product types and software environments.

B. Technological Trends by Product Type

103. It has been mentioned that computer software can be classified by product usage type into two broad groups: systems software and applications software. These have been widely used in conventional data processing, but have evolved considerably in their technology recently. Another class of software, integrated systems software, is recently emerging. This is directly associated with emergence of distributed computing, and represents the state of the art in software technology.

(1) Systems Software

104. A group of the programs required to make optimum use of the computer and its peripherals is referred to systems software. It includes operating systems which regulate and monitor the implementation of various programs. Next come compilers and interpreters which translate programs written in the computer languages such as Cobol, Fortran, Pascal, etc. into sets of machine-interpretable instructions. Systems software also includes data-base management systems (DBMS) which manage data files or individual data stored in the memory, controlling access according to the needs of running programs, and utility programs and debugging aids which are used for general housekeeping tasks such as copying files from one medium to another and program development testing.

105. Systems software is a must-to-be software to the computer systems. Its market has increased continuously as the hardware market expands. (see Table 14) However, systems software has been developed mostly by hardware manufacturers until recently. With more active involvement of independent software houses, many new systems software products are appearing.

Table 14. US Systems Software Consumption

$ million

	1985	1986	1987
Compilers & assemblers	856	932	1,025
DBMS	1,486	1,749	2,098
Diagnostic & debugging	871	1,060	1,314
Operating systems	3,864	4,688	5,678

Note: The figures for 1987 are "estimated".
Source: ADB/Electronics, Jan. 8, 1987.

106. There are new types of systems software recently appearing on the market as a result of progress in software technology: programming environments and programming generators. A programming environment is a programming aid system including various development tools, some general some specific to the field of application concerned. It includes fourth generation languages such as Ramis, Focus, and data-base management functions such as Mantis, UFO, IDMS, and even some advances in operating systems functions such as Unix as well as advances in conventional programming languages.

107. The programming generators may be included in programming environments as its sub-category. A program generator is supposed to convert a system definition written in a system-dependent notation into a program in the source form of a conventional programming language. Classes of program generators are report writers, macro generators, and structured language facilities.

108. So far, technological advances in this area have been achieved by major hardware manufacturers. Although they still dominate the development activities, the role of independent software houses is increasing.

(2) Applications Software

109. The most important contribution of computers to the user is its capability to memorize and handle extensive information very efficiently. Applications programs utilize these principles to solve various user's own problems.

110. There is a great variety of applications software, corresponding to the wide range of possibilities for applying computer processing. The International Software Directory classifies 107 groups, under five major headings: Accounting, administration, production, distribution; Banking; Design, modelling, simulation,

statistics;Insurance; Others. Most applications software consists of programs defining a modelled structure intended to explore the pathways defined within the structure.

111. Along with the development in hardware in the direction of faster speeds, higher memory capacities of in particular microcomputers, recent advances in software technology, have generated new applications software: decision-aid systems and expert systems among others.

112. A decision-aid system enables the user directly to solve some of his assessment and forecasting problems. Some programs evaluate various scenarios with a different set of assumptions, mainly in the financial field. The products such as IFPS, Stratagem, FCS, Express, Empire belong to this class.

113. Expert systems use AI (artificial intelligence) techniques.[9] They are designed to simulate the intellectual processes of an expert in a particular discipline, and to infer appropriate solutions to different situations. The expert system, though still very young, is highly promising in many professional fields. Several products are already on the market and widely used. Examples are Mycin, Prospecto, Litho, Decos, etc.

114. One technological variant of certain applications software is to use a custom hardware configuration in the case of a dedicated, permanent and large scale application, i.e., library and hospital management, airline reservations, various banking systems, and so on. This type of custom hardware/software configuration is called a vertical turnkey system. Its market is also growing rapidly.

(3) Integrated Software Systems

115. The most visible technological trend for the 1980s is the emergence of integrated software systems. They are designed for integrating different systems and application software components, sharing data and transferring results among the various programs on just a particular site or on a number of sites communicating each other. Integrated software systems for the latter purpose are especially important in distributed computing, a recent trend.

116. A new application in mainframes is to integrate many other applications programs with which it shares a centralized data base. Communication among applications programs relies on distributed architecture. Utilizing this approach, several software companies specializing in this area have recently launched many products such as IDMS, Focus, Fourth Generation System, Nomad, TIS, etc.

---

[9] AI will be discussed in detail in the next section.

117. This approach is also adopted successfully in applications software for microcomputers due to the improvement in micro-computer memory capacities. It satisfies the typical end-user who wants to develop an all-purpose data base and to transfer results, texts and graphics among different programs. Numerous products in this class are on the market. A partial list of them includes Wordstar, Perfectlink, Perfectwriter, Windows, Topview, Framework, Symphony, among many others.

118. The market for software integrating mainframes and microcomputers is emerging with the rapid spread of micro-computers in large organizations which communicate and share computer resources among various business units. These systems are still embryonic. Rather than supporting genuinely distributed software systems, their aim is to enable messages, data and programs to be transferred between the central mainframe and peripherals. Technical advances are gradually changing the role of the micro-computer from a simple terminal to a decentralized processor. The most advanced systems have been developed in highly decentralized large organizations. But certain standard interfacing products, such as Executive Peachpak II, PC Link, and PC-204 are also appearing on the market.

C. Changing Software Production Environment

119. The information technology industry, including the computer software industry, is expanding with rapid technological advances. Here is a presentation on the technological trends which seem to be the most significant to the information technology industries, and to computer software development.

(1) Computer Aided Software Engineering

120. A significant productivity increase, is being realized in engineering, design and drafting with Computer Aided Design and Drafting and Computer Aided Engineering (CAD/CAE). The very same logic is applied to automate computer programming with Computer Aided Software Engineering (CASE). Some CASE products are now appearing on the market. (see Table 15)

121. Since the beginning of computer programming, computer programmers have tired of repeating the exact same thing that have been done numerous times before. All programmers have a collection of subroutines that they themselves have programmed or have shared with other programmers. One of the objectives of CASE is to have an electronic mentor with a library of many subroutines in its portfolio. Once it is in practical uses, its impact will be enormous on the productivity in software development. Programs developed under CASE will be more accurate, contain more and better error traps, and reduce software development time by at least a half.

Table 15. Independent Program Design/Development Software Market

$ million

	1982	1983	1984
Large-scale	57	83	126
Medium-scale	68	121	198
Micros	5	11	26
Total	130	215	350

Note: The amounts are worldwide sales by US suppliers.
Source: OECD/IDC, 1985.

122. Despite the development of CASE, its impact on the applications backlog will not be significant. CASE is likely to be the most powerful in the programming stage. However, at least half of all programming time is for maintaining existing programs, or dealing with user problems with existing programs. Unless all of application programs are rewritten using CASE technology, CASE is of little value in maintenance of existing programs.

(2) Artificial Intelligence

123. One of the hottest areas of development in computer software is applications of artificial intelligence (AI) technologies. AI technologies may be applied in various ways. A major use of AI is expert systems. An expert system is to program the knowledge of human expert into a computerized data structure, along with rules of inference for the application of that knowledge. AI expert systems attempt to teach computers how to make decisions based on probabilities and learned experiences.

124. Another field is the investigation of ways in which a computer can be taught to mimic the patterns of human learning, reaction, and thought. This area is a long way from practical applications. Great effort is being made on AI to develop the technology and apply it to a wide variety of applications ranging from medical diagnosis to weather forecast.

125. AI applications requiring great amounts of information processing will be promoted by the on-going advances in computer technology. One promising approach is the development of Neural Networks. Neural networks are an advanced variation of parallel processing computer architecture. Neural networks are supposed to mimic the human brain by having millions of parallel processes going on at the same time.

(3) Network Architecure Standards

126. The applications of distributed and integrated computing require network architectures. The International Organization for Standardization (ISO) has developed a system network architecture standards for the orderly development of networks using heterogeneous network components. The reference model, OSI (Open Systems Interconnection) has been approved as an international standard. This reference model provides an architectural description whereby systems are decomposed into seven layers according to the services to be performed.

127. In the field of wide area networks (WAN), several ISO and CCITT (Consultative Committee on International Telephone and Telegraph) standards exist to facilitate interconnection. The IEEE 802 Committee has also developed widely accepted standards in local area networks (LAN). Several advanced countries are now encouraging manufacturers to develop products in conformance to the reference model.

128. Due to strong pressure from the user community, the major manufactures are increasing their role in the establishment of the standards on the Open Systems Interconnection. It is likely that their product lines and proprietary architectures will evolve towards compatibility with the international standards.

(4) Japanese Fifth Generation Computer Project

129. In order to develop a radically new type of computing based on the most recent technological advances in information technology, Japan has launched a project in 1980 with an estimated budget of $500 million over ten years, under the auspices of the Ministry of International Trade and Industry (MITI) with leading Japanese manufacturers contributing scientific staff.

130. The terminal goal of the project was not clearly set up in advance, but their intention was to develop a very powerful super-computer running a great many programs simultaneously or a machine that processes not only individual data but also formalized knowledge.

131. At this time, it is very unlikely that the Japanese can build such a machine before the end of the project period. Through the project, however, significant advances in several areas of information technology are expected, including voice recognition, voice synthesis, computer memory and memory management, artificial intelligence and the parallel processing. Whatever the outcomes are, development of software will be affected by advances in information technology through such a projects.

(5) IBM's OS/2

132. In 1987, IBM announced the PS/2 family (personal system 2), the successor to the popular IBM-PC. The PS/2 line consists of two 8086 based computers, several 80286 based computers, and two 80386 computers. Although all PS/2 computers run existing operating systems, such as PC-DOS and MS-DOS, a new operating system called OS/2, was announced at the same time as the PC/2 computers. OS/2 works only on 80286 and 80386 members of the PS/2 family.

133. OS/2 is a family of operating systems with the capability of multiple processing. This is a big step, since current PC's are only able to perform one task at a time without special systems software, and current multi-task system software products have definite limitations.

134. The impact that OS/2 operating system software will have on computer software development is significant. Considering IBM's marketing power, the OS/2 operating system will soon take a big share. Thus, successful marketing of a program is likely to depend on its capability on OS/2. Furthermore, exploiting the advanced features of OS/2, software developers may add new products to their product line.

(6) Unix Operating System

135. Unix is a powerful operating system developed by AT&T. There are increasing trends in its use, which may have significant impact on nearly all aspects of information technology, especially software development. The Unix operating system has its potential for becoming a universally used and accepted operating system. The ability of Unix to run on almost any computer, and its inherent communications capabilities are the sources which give it such potential.

136. Unix is steadily gaining its market share. (see Table 16) In a recent press release, AT&T announced a version of Unix for the Intel 80386 CPU which means that very powerful microcomputers, including the new IBM P/S-80, will have a Unix operating system that can take advantage of the capabilities of the 80386 chip and the universe of software written for Unix.

137. There is another interesting development in the Unix operating system. A generic version of Unix called Posix is being defined by ISO. This generic "standard" definition will probably open the Unix marketplace to a larger number of users. As more use the Unix-based machine, more software running on the machine will be demanded. Therefore, it is inevitable for software houses to write Unix-based programs, and this requires more complete control on the Unix environments.

Table 16. Forecast of Global Unix Machine Shipments

thousand units

Year	Shipments
1985	100
1986	148
1987	219
1988	324
1989	480
1990	700

Source: Computerworld Focus, Aug. 20, 1986

(7) Technical Progress in Hardware

138. Increases in computing capacity make applications which were previously impractical, practical. Thereby, the areas that computers can be used will expand, and software developers will write more diverse programs that take advantage of the new capabilities to meet these market needs. Greater power at lower cost will result in more organizations to make use of information technology for business, and more capabilities per dollar will result in more work being done with computers.

139. Since graphics, voice synthesis, and voice recognition are demanding of high speed and large memory of computers, new technological advances might enable computers to be used for speech recognition, spoken word transcription, real-time translation between spoken languages, and speaking robots doing household chores. For the most part, we expect software to be larger and more sophisticated technologically as computers with more greater capacity become available.

140. Software development in future will be on this way through hardware advances such as VLSI, large capacity microprocessors, bio-chips, optical discs and fibers, superconductors and Josephson Junctions, by improving the scope for both local processing and distributed computing. The leading companies in hardware technology are now working vigorously in this direction. Furthermore, a totally new approach on computing, parallel architecture computers, is under study. That is expected to replace in future the current basic computer technology, the von Neumann principle of sequential programming, as has already been heralded for several years. The probable consequence would be a further technological reorganization on the software side.

Chapter 4

ACTIVITIES OF SOFTWARE INDUSTRY IN DEVELOPED COUNTRIES

141. In spite of macro-economic difficulties, information technology expenditures have continued to grow throughout the advanced countries, and software developments stand out as one of the industries growing most vigorously at presen

142. The progress of the software industry cannot be achieved only by the autonomous efforts of the firms concerned. It calls for a range of external factors: new scientific and technical knowledge; new professional skills; diverse infrastructures; appropriate standards; and the financial and legal frameworks. Governments play an important role in the availability of these factors towards development of the software industry.

A. Research and Development Activities

143. Since technological achievements in software can be disseminated at extremely low cost, and is afforded practically no legal protection, software research is distinctive in the public good character of its advances unlike most other technologies.

144. In this respect, private investors are of relatively little interest in basic software research. This accordingly justifies the responsibility of governments for the R&D organization, funding and orientation of software research.

145. University research centers, mainly within computer science departments and specialized laboratories are the most prominent, and main components of software research in advanced countries.[10] Many of these centers have independent status and can undertake large-scale research programmes for development of software technologies. Particularly in the United States, defence-oriented software

---

[10] The examples are the Rand Corporation and Information Sciences Institute in the United States, National Physical Laboratory and Royal Signals and Radar Establishment in the United Kingdom, Centre for Electronic Calculation in Spain, Institut National de la Recherche en Informatique et Automatisme(INRIA) in France, Gesellschaft fur Mathematik und Datenverarbeitung(GMD) in Germany, the Institute for Computer Technology(ICOT) in Japan.

research has originated many advances in the field under the guidance of various agencies.

146. Examples include Cobol by the Navy in the 1950s, most of the initial software engineering research sponsored by the Air Force and NASA in the 1960s, Ada by DOD among others. The Defense Advanced Research Projects Agency (DARPA) has been a major contributor to computer research in time-sharing, networking, distributed computing, parallel processing and special purpose architectures, and artificial intelligence.

147. A Few of the leading hardware manufacturers also maintain fundamental research capabilities in the information sciences.[11] In addition, various cooperative research programmes are also undertaken in the advanced countries under the government budget support. ESPRIT in the European Community, the Alvey Project in the United Kingdom, EPICEA in France, and the Japanese Sigma project are the examples. In the programmes, governments play a catalytic role in both financial assistance and programme formulation.

148. The main problem facing the countries is a shortage of human resources available for research, especially young researchers at the Ph.D level. Fewer and fewer of them choose university careers, as most prefer positions in industry. An NSF survey in the United States has shown that these young professionals are motivated mostly by the working environment they are offered. Financial rewards are a relatively secondary consideration for researchers and research managers in this area.

B. Manpower Training

149. One of the main policy problems in most developed countries is the gap between supply and demand for qualified human resources in information technology. (see Figure 4) In view of the present trends in technological developments, this gap is getting wider and wider throughout the various types of software specialists. According to an estimate, the shortfall is between 50,000 and 100,000 software professionals at present, and it will be between 850,000 and 1 million by 1990 if no action is taken.[12] In Japan, according to MITI estimates, the demand for software engineers will

---

[11] IBM's Yorktown Heights, San Jose and Zurich (Switzerland) laboratories, AT&T's Murray Hill Laboratories, Digital Equipment's Scientific Research Center in Palo Alto and Western Research Laboratory in Los Altos, Hewlett-Packard's Computing Research Laboratory in Santa Clara among others, contributed to many advances in software technology.

[12] OECD, 1985.

rate of 11 per cent per annum, while the supply will grow from 43,000 in 1985 to 1,180,000 in 2000, the resulting lack of software engineers amounting to 970,000 in 2000, failing a vigorous automation of software production.[13]

150. Software professionals can be classified into three groups based on their training types: technicians, software engineers, and in-service trained specialists. Technicians, mainly covering analyst programmers and application programmers, qualifies with an average of one or two years training after the baccalaureate or equivalent. Training is provided in public and private institutions and comprises a wide range of professional qualifications, varying in level and quality, with very little standardization. Software engineers are graduates from a university computer science department or schools specializing exclusively in software engineering. In-service training is mostly provided by specialist service firms or by the employer firm. This covers a wide range of types and quality of training, and plays a particularly important role in the industry. In France, for example, computer service companies allocate an average of 6 per cent of their total wage bill to in-service training for staff. Japan stands out as the country with the most elaborate in-service training systems, most firmly established as part of a professional's career.[14]

151. Governments in most advanced countries are now preparing active policies to make up the shortfall in software specialists. However, they also face various difficulties in implementation of the training policies.

152. Teaching personnel are very scarce at higher as well as secondary level, in both public and private sector alike. Various educational organizations are trying to devise special arrangements to palliate their instructor shortfall, i.e., engaging specialists in industry part-time, retraining instructors from other disciplines, etc.

153. The problem in material resources is mainly how to finance the purchase and maintenance of computer equipment by educational establishments. Falling hardware prices and the emergence of microcomputers are partly alleviating this problem, though it is still a formidable obstacle. In some cases, hardware manufacturers are allowing considerable discounts or donating the equipment to educational establishments, especially in the United States. But this may have the disadvantage of introducing commercial bias, since manufacturers are mainly interested in training future customers to use their own equipment.

154. Another difficulty in training policy comes from the rapid progress

---

[13] IPA/MITI, 1988.

[14] OECD, Software: Emerging Industry , 1985

in information technologies, and thus continuously changing software development environment. There are no longer any standard environments such as Fortran, Cobol, IBM 360 or equivalents in the computer world to serve as a universal reference for training programmes. This implies that those responsible for educational policies in the public or private sectors must be able to reassess the state of the art technology so that training institutions can be continuously adapted accordingly. But this is a difficult and expensive task. There are many government or quasi-government commissions, and trade associations in advanced countries to monitor developments and make recommendations on these issues.

155. Today, most governments adopt decentralized training policies, in which public sector and independent private training bodies make their own decisions about training programmes and courses, reflecting demand from industry. There are some advantages in the approach. Especially it allows the necessary flexibility to the training bodies in the situation of staff shortages and technological uncertainties. But it raises an issue - diverse qualification of trainees.

156. In certain countries, governments have established common qualification standards conferred by competent training agencies.[15] An approach used in Germany, and in Japan, and to some extent in the United States, is to apply country-wide examination procedures for qualifications attested by national certificates: in Germany the Certificate for Economic Information Engineers; the MITI's electronic technician and software engineering diplomas in Japan; the Certificates in Data Processing in the United States awarded by ICCP (Institute for Certification of Computer Professionals), a private association.

C. Standardization

157. Standards play a vital role in the efficiency and productivity of software development. Their importance is growing with the introduction of integrated and distributed computer and communications systems involving a great number of hardware/software interfaces. The build-up of integrated information systems, the division of labor and effective competition among software companies, software portability between different systems and different sites, the transferability and interchangeability of

---

[15] OECD(1985) introduces many examples : Assistant Engineer for Data Processing, Mathematical-Technical Assistant, Economic Assistant for Information Science, Technical Assistant for Information Science and Certified Information Engineer in Germany; EDP-assistant, Datanom and Datalogists in Denmark; Datanomists and Systems Analysts in Finland, Computer Experts and Accountant Programmers in Italy; Program Analysts and Computer Analysts in Switzerlands.

professional resources, and more broadly, open markets and the numerous potential economies of scale and of learning in this field all depend on the existence of standards.

158. Standardization issues in hardware systems are associated with the acceleration of technical innovations which brings risks of incompatibility between various components of systems if they are not standardized. The increasing trend of international trade also requires internationally acceptable standards.

159. Most of the standards applied in information technology at present stem from those of a few hardware manufacturers, who hold the dominant position in hardware market. However, a growing need is developing for generally accepted standards, involving the wide participation and co-operation of large and small hardware and software producers and end users.

160. It is difficult to organize and implement this kind of standardization nationally, and even more so internationally, because of the numerous commercial interests involved and rapid technical progress which leads to the frequent launching of innovations on the market. Nonetheless, substantial progress has been made, notably in the validation of new versions of programming languages and especially with the gradual acceptance of the multi-layer network architecture OSI (Open System Interconnection) developed by the International Organization for Standardization.

D. Government Support for Software Companies

161. We have noted that service firms specializing in software must overcome certain difficulties inherent from the industrial characteristics to enter the market and grow. Difficulties have been found to stem from the ill-defined economic and legal status of software. Government endeavors to specify the economic and legal status of software and to introduce appropriate accountancy, tax and legal frameworks are consequently of great importance to the activity and growth of software houses.

162. Software houses are also facing financing difficulties, because of the increasing outlet in development and marketing costs due to accelerating progress in software technology and aggravating market competition. Financial organizations are reluctant to finance such companies, whose assets and output are intangible and volatile.

163. Government authorities usually support software companies by intervening in several ways. Support measures for investment, R&D promotion such as tax concessions, grants, accelerated depreciation, etc., today in force in most advanced countries, do not automatically apply to software firms whose investments and output are intangible. Governments are revising their legislation to include software products or adding an ad hoc clause for software.

It is also noticeable that many software companies locate in technology parks established by governments or local authorities in many countries. Companies in the park cooperate in software production and marketing. They may share costly facilities.

164. Only a few governments have so far introduced support programmes specifically for the software industry. The most elaborate programme has been implemented by the United Kingdom, the Software Products Scheme. Under this scheme, the development of software packages is subsidized by grants of up to one third of development costs. Grants are offered only to innovative projects. To get a grant, applicants must demonstrate technical and commercial competence, as well as the need for assistance. Since 1983, some 215 projects have been assisted and grants of 29 to 39 million pound have been made, inducing investment of 108 million pounds.[16]

165. Japan is also operating a programme to support development investments by software companies. The Information Technology Promotion Agency (IPA) funds up to 100 per cent of the costs of developing software which it considers to be marketable. In 1984, 33 projects were supported, with a total budget of 3.5 billion yens. The IPA also guarantees loans from the long-term credit banks to software companies. It is reported that it backed 1519 credit arrangements amounting 58.4 billion yens between 1970 and 1985. Another major support measure for software companies in Japan is a special tax arrangement, under which 50 per cent of earnings from packaged software sales can be paid into a development fund exempt from taxation for four years.

166. It is inevitable for a software company to aim at the international market in order to overtake what may be an unduly narrow home market. But most find it difficult, because of financial, organizational and cultural (mostly linguistic) reasons which loom large in this industry. Government export measures, as a result, may be supporting software companies.

167. Support can take various forms, sometimes linked to information technologies such as the UK Government's IT export scheme. There may be various subsidies to exports, special foreign trade services, support for missions and trade fairs, insurance for export projects. Many software companies have taken advantage of these measures even in the advanced countries.

E. Activities in the United States and Japan

(1) The United States

---

[16] OECD, 1985.

168. The United States occupies approximately 75 percent of the global market for computer software today. But, it is very likely the position will be eroded significantly in near future, because of vigorous activities in other countries. To keep her position in information technology, large amounts of research funds are granted every year to educational institutions for projects relating to computer hardware and software. Every year various departments within the Government invest billions of dollars in computerized systems designed to accomplish objectives ranging from environmental monitoring to national defense.

(a) Education

169. Development of computer software is now a hot issue within business, government, and educational organizations in the United States. Nearly all university and junior college curricula in US include one or more courses on Computer Science.

170. However, American educational institutions fail to meet the skills or training required by the data processing industry. The problem stems from mainly the shortage of qualified instructors. Teachers are frequently university graduates with little industry experience. Furthermore, teachers with the greatest skills and abilities are frequently recruited away from teaching by better working environments offered by business that universities cannot match. This situation is steadily improving, as more expenditures are spent in information technology education.

(b) Tax Policy

171. Although there have been great advances in Computer Science from American universities, the equivalent advances in software and hardware technology have been at the hands of entrepreneurs. Until the end of 1986, there were significant tax advantages associated with all areas of research and development(R&D) including computer software development.

172. However, the tax code revision of 1986 wiped out what had been a significant tax break for software development efforts of American companies. The current USA Administration seems to bend on continuing its policy of benign neglect and has left the developments in the computer software industry to the whims of the free markets.

(c) Copyrights

173. The US government has extended the protection of copyrights to computer software. Through the combination of legislative revisions to the USA Copyright Act, and numerous Supreme Court decisions,

copyright protection is gradually growing with respect to computer software.

174. In general, copyright protection is granted not to the idea, but to the expression of an idea. Much of the evolution of US copyright law with respect to computer software has been defaulted to the courts, as the US Congress fails to incorporate many of the subtlety of computer software into copyright legislation.

175. There is still substantial work that needs to be done to define legal protection for computer software. Legislative work continues, as does the lengthy process of litigation to obtain High Court rulings. The USA, by virtue of its position as the leading producer and consumer of computer software, will continue to be the focus of worldwide attention in the evolution of legal protection for computer software.

(d) Government Consumption

176. Agencies of the US government allocate huge budgets for computer software and programming services. As the world's largest consumer of computer software and programming services, the US government has a significant impact on the computer software development. Not only is the government procurement a good market, but also technologies developed on government contracts are diffused into the software industry so as to improve the productivity of commercial software product development.

177. The Department of Defense (DOD) spends about US $5 billion per year on computer software development. In an attempt to optimize these expenditures, the DOD has mandated that all programming on DOD projects shall be performed in the programming language Ada, which has been developed early in the 1980s. To some extent, Ada helps standardization of the computer languages.

178. In sum, despite the absence of any noticeable software policy on the part of the Government, the United States still accounts for 75 per cent of the global market. As more and more organizations introduce computerization, and advances in information technology speed up, the United States will continue to play a central role in software market and technology development.

(2) Japan

179. Japan's Ministry of International Trade and Industry (MITI) has actively encouraged software development for many years. As a result, Japan's software market has grown from less than $2 billion in 1982 to over $5 billion in 1986. Now with over 1,000 companies,

the Japanese software industry is the second to US.[17]

180. Japan, like the rest of the developed countries, is experiencing a growth of the application backlog and a shortage of skilled programmers, as custom software's market share has shifted from 90 per cent to 60 per cent in favor of off-the-shelf package programs.

181. Promotional activities taken by Japanese government have already been presented briefly in the previous section. In the following, major ongoing endeavors are introduced.

(a) The Fifth Generation Computer Program

182. The ambitious program to develop a fifth generation computing technology of hardware and software has been started under the initiative of the Government. As discussed previously, few people expects the project to succeed as initially planned. However, the project awakened the Japanese competitive spirit, and the technologies spun off from the intense research will continue to surface for the next ten to twenty years.

(b) Sigma

183. Sigma (the Software Industrialized Generator & Maintenance Aids) project aims to develop a set of advanced software engineering tools for promoting a high level of automation in software production in Japan, and to develop a nationwide on-line database of software tools based on the Unix operating system. The project is carried out by the co-operation of IPA (Information Technology Promotion Agency) and all major Japanese computer makers and software companies. The 1985 budget for the project was 3 billion yens, and total amount of 30 billion yens is envisaged by 1990.

(c) Tron

184. Tron (The Real Time Operating System Neucleus) is an effort by Japan's big computer firms to develop a totally new operating system with complete universality. The ambitious project, coordinated by IPA, consists of several sub-projects: I(Industrial)Tron for developing high speed processors; B(Business)Tron, a man-machine interfacing OS for standard microcomputers in the 1990s; C(Central)Tron for efficient equipment control; and M(Macro)Tron for integrating all the other Trons. At present, ITron is under development by leading Japanese companies including JEC, Matsushita, Hitachi, etc. Success of Tron would reduce current reliance on operating systems from IBM and AT&T, increasing independence in this area of software. Furthermore, it will open a totally new world in the computer environment.

---

[17] An unauthorized report estimates around 2,000 companies.

Chapter 5

THE KOREAN SOFTWARE INDUSTRY

185. Software in Korea has developed as ancillary to hardware until recently. Hardware manufacturers, through copying foreign software packages or developing internally or by contract with independent software house provided software along with their hardware. This is a similar pattern observed in advanced countries in the early stage of their software industry. In those countries, however, the rapid increase of the software demand since the mid 1970s has attracted many new entrants to the software industry, and thus the industry started to take a solid foot on the ground.

186. The situation is different in Korea in that software demand never exceeded the supply capacity of the software houses. No backlog has been observed since beginning of its activity. As of the end of August, 1987, 392 firms were competing in the meager local software market.[18] This implies Korean software companies are not aiming at meeting the current demand, but based on the expectations of the future growth. Approximately 50 companies join the market every year, and almost equivalent number of companies leave the market. This chapter deals with the recent trend of and major problems facing the Korean software industry, and Government efforts to promote the industry.

A. Characteristics of the Korean Software Industry

(1) Building up Computer Systems for Korean

187. Basically, computers have been designed for using mainly English. English is the most widely used computer language. For the full-fledged local software market, a computer operating system based on Korean is of utmost importance, and the first step to be taken.

188. Unfortunately, developing a code system for Korean workable on existing computers is highly intricate because of its linguistic characteristics. Two major approaches are currently applied for incorporating Korean into the computer system: one approach is through modifying hardware by a new board design, and the other is through modifying software using a graphic board. In either way,

---

[18] The number is only the companies registered to MOST. Around 1,000 are expected in the software business.

Korean programs should be written in accordance with a given hardware configuration. Therefore, compatibility of software for Korean is very low across different hardware configurations.

189. For more complete Koreanization of computer systems, the operating system itself should be written in Korean which requires a set of codes for Korean. Because the code system for Korean is different across computer hardware manufacturers, software developed by them has different structures. These complicated nature of dealing with Korean is the major obstacle to the development of software in Korea.

(2) Changes in the Software Demand/Supply Pattern

190. Recent trends show a change in Korean software market. Systems software, which used to be supplied by hardware manufacturers, is increasingly being supplied by software houses and software distributors. This is due in part to the increasingly costly and time-consuming in-house development, and in part to the overall improvement of quality, reliability and productivity of the independent software houses.

191. In case of applications software, the greater part of software demand in Korea was for building MIS in business organizations. Office systems vary among different users, and consequently there used to be some advantages of supplying necessary software through in-house development. Thus, demand for standardized packages was very few and most of applications software have been developed by in-house development teams.

192. Though this is still the case, there developed some factors to change this kind of practice. Office standardization, increasing maintenance costs of in-house software development teams, time-consuming process of internal development, and increasing productivity of independent software houses have contributed to the increasing utilization of standardized packages.

(3) Improving Quality of Software

193. Because of the low technical level of the local software houses, most computer users in Korea develop their own software needs in-house, approximately 55 per cent, compared to the purchase of commercial packaged software and outside contracts (see Table 17). The large share of in-house development of software in the total software supply hindered the establishment of the software market, and eventually delayed the overall development of software industry.

Table 17. Software Procurement Sources by Korean Users

per cent

	Systems software	Applications software	Total
In-house developments	12.4	42.6	55.0
Contract developments	6.9	11.0	17.9
Lease and Purchases	23.7	3.4	26.1
Total	43.0	57.0	100.0

Note: This table is made based on survey data, as of Oct. 31 1987.
Source: KIIA, Survey on the Software Procurement and Maintenance, Korea, 1987.

194. However, improvement in the quality of local software and in capability of local software houses change the procurement pattern of computer users to utilize outside software vendors more. The quality improvement of software developed locally has been achieved mainly through increasing the size of software firms, and increasing technological license agreements with foreign firms, overall advances in information technology and, above all, continuous endeavor of software companies themselves for it.

195. Specialization of software houses is also very important for improvement of reliability and quality of software. Though the specialization in the software industry is still very low, chances are very high since the software markets are growing in various areas.

(4) Short Supply of General-Purpose Software

196. Though there is a fairly large demand for general-purpose software packages in Korea, the current supply by local software houses is very limited. This is mainly due to the variety of user office systems. Business organizations in Korea have different systems. Above all, most needy general-purpose packages are as follows: software for on-line screen design; software for report generation; PC software; telecommunications software for interfacing hardware.

(5) Little Maintenance Activity

197. The reliability of software largely depends upon the maintenance

activities. However, maintenance activities are hardly observed in the local software industry. There are many factors explaining the situation. Technologically, the local companies do not have capabilities to handle complicated control programs such as IMS, MVS/SP, CICS, NCP. Most of them have little experience in handling program errors in software. They really could not build up know-how's in quality control and maintenance of software, because of their short history. Furthermore, insufficient documentation and standardization make maintenance activities by the local companies difficult.

(6) Trade Imbalance

198. Software export by Korea amounted to $3.90 million in 1984, $7.2 million in 1987. But, imports during the period triples the amounts. (see Table 18) The growth rate of export, 22 per cent during the period, is especially low, compared to the other industries. The share of export in total software production is less than 10 per cent in 1985. This implies software export is now only at the formative stage. Furthermore, software export consists mostly of low-level software services such as key-punching and coding for foreign software houses.

Table 18. Software Trade in Korea

$ million

	1984	1985	1986	1987
Export	3.9	5.1	6.3	7.2
Import	11.9	23.8	22.4	29.1

Source: FKII, Survey of Import-Export Trends of Information Industry, 1988; OCA, Trade Statistics, various issues.

(7) Small Software Houses

199. Korean software industry is characterized by the severe local competition of small software houses with very low level of specialization. Firms with paid-up capital less than 100 million Won consisted of 58.2 per cent in 1987. Almost half of the companies have less than 15 employees. (see Table 19 and 20) The small size of companies may be a characteristic of software industry. However, it is very likely that larger firms are on a better position for enhancing competitiveness.

200. The growing market for large government projects such as the computer information network for administration, has recently induced a number of larger firms to join the industry. Larger firms, confronting increasing uncertainties in their own business environment, begin to take a greater stake in computer industry for the future. Their strategy in software business is rather far-sighted. At first they develop software just for their own uses and the local market, and then expand to the export market by increasing their technical capability in this area through technical tie-up with foreign firms.

Table 19. Distribution of Korean Software Companies by Paid-up Capital Size

	Less than 100 million Won	From 100 million to 1 billion Won	Over 1 billion Won	Total
1985	176 (57.7)	91 (29.8)	38 (12.5)	305 (100)
1986	200 (56.8)	109 (31.0)	43 (12.2)	352 (100)
1987	228 (58.2)	116 (29.6)	48 (12.2)	392 (100)

Source: MOST, Guidebook of Korean Software Industry, Korea, 1986; KIIA (Korea Information Industry Association), Survey on the Actual Conditions of Information Industry, Korea, 1987.

Table 20. Distribution of Korean Software Firms by Employment Size (1987)

Employees	Software companies
Less than 15	190 (48.5)
16 - 30	74 (18.8)
31 - 50	36 (9.2)
51 - 100	29 (7.4)
More than 100	63 (16.1)
Total	392 (100.)

Note: Numbers in ( ) are the percentages.
Source: KIIA, Survey on the Actual Conditions of Information Industry, Korea, 1987.

(8) Low Level of Specialization

201. Though there are increasing number of firms specializing development of software, 22.4 per cent in 1986, the majority pursue software as a side business of hardware. (see Table 21) This implies that the Korean software industry is still embryonic, and not separated from hardware industry to stand alone. Due to the smallness of firms and the dependence upon hardware, even the import of the up-to-dated technology in this area is very limited.

Table 21. Distribution of Software Companies by Business Scope

	Companies carrying both s/w and h/w	Companies carrying software only	Total
1985	252 (82.6)	53 (17.4)	305 (100)
1986	273 (77.6)	79 (22.4)	352 (100)

Source: MOST, Korea, 1987.

B. Software Supply/Demand in Korea

(1) Overview of Korean Software Market

202. In order to understand the software industry, figures on business activity is essential. Because of no official statistics available on the Korean software market, and its instability partially due to the reluctance of the software houses to announce their past business activities, no precise estimation is possible at this point. However, local software demand is roughly estimated to be $76 million in 1985, growing with an annual rate of 51.9 per cent since 1983.[19] (see Table 22)

203. While the software demand increased significantly, the portion of the local supply, defined as total production less export in the local demand, declined from 82.0 per cent in 1983 to 68.6 per cent in 1985. On the other hand, software import has increased rapidly from $5.9 million in 1983 to $23.8 million in 1985. This rapid increase of software import has resulted from the wide diffusion of

[19] The local demand was derived from the total supply less export. The figures are different from KIET's report because of different sources used.

computers and increasing sophistication of user needs. Because of the low technological level of the local software houses, domestic demand for highly sophisticated software should be met inevitably by foreign suppliers.

Table 22. Supply/Demand of Software in Korea

$ million

	1983	1984	1985	Growth rate
Supply				
Local production	30.8	40.0	57.0	36.0 %
Import	5.9	11.9	23.8	100.8 %
Demand				
Local demand	32.8	48.0	75.7	51.9 %
Export	3.9	3.9	5.1	14.4 %

Note: Local production does not include in-house development
Source: KIET/EIAK, Information Industry Yearbook, 1986;Office of Customs Administration, Various issues of Trade Statistics.

204. The above figures on supply and demand does not include in-house developments. The in-house development in Korea is estimated even greater. According to an estimate, in-house development of software amounted $52 million in 1983 and increased to $85 million in 1985 with an annual growth rate of 27.9 per cent. The figures are compared with commercial software development of $30.8 million, and $57 million, respectively. (see Table 23).

Table 23. Software Development in Korea

$ million

	1983	1984	1985	Growth rate
In-house development by users	52.0	60	85	27.9 %
Production by software vendors	30.8	40	57	36.0 %
Total	82.8	100	142	31.0 %

Source: KIET, The Structure and Growth of Software Industry, korea, 1987.

205. This high proportion of in-house software development by users is closely related to the low technical capability level of local software houses by which very particular and specialized needs of users can hardly be satisfied by local firms. It is, however, observed that with the technological development of Korean software industry the proportion of in-house development to the total development gradually decline, and more users find software for their uses in the market place.

(2) Demand/Supply Structure

206. The rapidly increasing trend of software import in Korea has already been mentioned. As a result, the rate of self-supply (ratio of the supply by local software companies to the total domestic demand) in 1983 was 82.0 per cent, but decreased to 68.6 per cent in 1985. The high rate of local self-supply in earlier years is due to the narrow user base of computers. Computers in Korea were used mainly for statistical analyses or routine management purposes. According to the survey of the Korea Information Industry Association (KIIA), now the Federation of Korean Information Industries (FKII), on the computer utilization in 1983 shows the proportion of computer usage for specialized jobs was less than 10 per cent. (see Table 24)

Table 24. Applied Areas of Computers in Korea (1983)

	Percentage
Management decision-making	1.7
Engineering	6.4
Advertising	0.4
General office management	81.6
Others	9.4

Source: KIET/KIIA, Survey on the Actual Conditions of Using Computer Systems, 1983.

207. The growing import dependence is due to the overall expansion of software market by increasing computer utilization on the one hand, and on the other hand to the low level of specialization and the low technical capability of local software houses which cannot meet the increasing demand for highly sophisticated specific applications software as computers are widely used in various organizations. Imported software consists mostly of specific-use programs and systems software. (see Table 25)

Table 25. Types of Imported Software by Korea (1983-1985)

$ million

	1984	1985	1986	1987
Applications software	3.6	11.9	7.5	11.9
System control software	5.7	7.7	8.3	10.3
Languages compilers	1.5	1.5	2.0	1.4
Utility programs	0.7	2.1	2.8	4.6
Not classified	0.4	0.6	1.8	0.8
Total	11.9	23.8	22.4	29.0

Source: FKII, Survey on Import-Export Trends of Information Industry, Korea, 1988.

208. To increase the technical capability of local firms so as to be competitive in the foreign market, as well as local, the utmost important is the specialization to specific field of software. Only a very handful of software houses out of total 352 firms enlisted to MOST as of October 1986 are specializing in a few limited areas.

209. Thus, in Korea a number of firms compete in a limited local market of general-purpose applications software. Such a situation has been resulted from the overall narrow local market. The local demand in a specific area is too small for a firm to survive through a specialization strategy.

(3) Prospects of Local Software Market

210. There are two contrasting views on the Korean software market growth. Some expect a rapid growth and the other expect a rather sluggish market. The former based their view on the past experience of local hardware market. It is also grounded upon the expectation that the international trend of rapid informatization will induce massive software demand. However, considering the low level of computer utilization in Korea, the expectation of rapid expansion of the local software market would be too naive. Furthermore, the long and complicated process of software development, and the slim opportunities for immediate market creation for a newly developed software product, except for a few general-use packages, also dilutes the expectation of a rapid market growth. Taking all these factors into consideration, a rather conservative forecast is made in this section.

211. The software market, including in-house development, is expected to grow from $160.7 million in 1985 to $390 million in 1988, and to $930 million in 1991. Commercial software market alone is expected to grow to $205 million in 1988 and to $582 million in 1991. (see Table 26). In 1985, in-house development was greater in amount than commercial production. This situation is expected to be reversed around 1988. The increasing trend of purchasing software outside is due to the improvement in reliability of externally supplied software and the increasing costs of keeping in-house computer departments. The share of in-house development will continuously decrease, compared to the purchase in the market.

Table 26. Forecasted Software Supply/Demand in Korea

$ million

	Forecast		Growth rates	
	1988	1991	1986-88	1989-91
Supply total	390	930	36.4%	33.6%
Vendors production	172	487	44.5%	41.5%
In-house Production	185	348	29.6%	23.4%
Import	68	245	59.7%	53.3%
Demand	390	930	36.4%	33.6%
External Purchase	205	582	44.0%	41.6%
In-house development	185	348	29.6%	23.4%
Export	35	150	90.0%	62.4%

Source: KIET, Structure and Growth of Software Industry, Korea, 1987.

212. In supply side, the software production by local software companies is expected to grow to $487 millions in 1991 from merely $57 million in 1985 with an annual growth rate of over 40 per cent. However, the increasing trend of software import will also continue until 1991. It is estimated to reach $245 million in 1991 with the annual growth rate of over 50 per cent.

213. The rapid increase of software import is closely related to the increase in hardware import. As long as the advanced computer architectures, such as supercomputers, artificial intelligence computers, or highly sophisticated mainframes, have to be imported, the related systems and applications software also have to be imported. In addition, imports of specific-use applications

software whose demand cannot be met by the local firms because of the technological complexity, will also increase significantly.

214. As a result of rapidly growing demand for imported software, the import dependence will increase from 24.3 per cent in 1985 to 42.1 per cent in 1991. The greater part of the local software market is supplied by imports.

C. Technology

(1) Overview

215. The technology development activities by local firms are concentrated mostly around learning and applications of existing technologies, while R&D on the frontier technology is only at the early stage. This is mainly due to the rather short history of computer utilization in Korean business and government organizations. They were just busy in using software provided by hardware manufacturers and modifying the foreign software for using Korean. Furthermore, the import of foreign technology is also concentrated on a few specific areas, which leads to un-balanced development. That is, compared with rather active development efforts in the area of applications software, the systems software has long been neglected.

216. More specifically, the level of software technology in Korea is quite low in every stages of software development. In case of systems software, most of operating systems software, DBMS and telecommunications software are imported. Minor modifications in operating systems such as UNIX, CP/M, MS-DOS for quality improvement, compatibility, real-time function, and embedding codes for processing Korean are all about. The basic technology of applications software is also at the infant stage. The data base management systems have long been introduced, but the local design capability is still very low. Software for CAD/CAM and software applying artificial intelligence is just beginning to develop. Even standardization for Korean is not well established.

217. But since 1980, the development of software industry has been seriously taken into account. Government and industry started to invest time and concern to this industry, and also to train professional manpower in this area. Owing to the concerted efforts of Government and industry, software development for practical uses is in view in the near future. The experience of local development of INS (Integrated Network Service) in the 1986 Asian Games established a sound base for development of computer systems for 1988 Seoul Olympic Games.

218. Though there is still significant dependence upon imported foreign

technology, the local firms are accumulating technology and experience of software development through participating in large government projects such as the national information network system projects. To understand the current status of local software technology, the local R&D activities, the level of technology, and the import of foreign technology will be reviewed in detail.

(2) Local R&D Activities

(a) Software Development by Local Software Houses

219. Although there is no accurate estimate available in Korea, according to EIAK (Electronic Industry Association of Korea) survey in the mid 1986, 93 firms developed 1,032 programs. Of them, 83 cases were reported in the area of systems software, 939 cases in the applications software. (see Table 27) This is a dramatic increase compared to the figures of the mid 1985 - 71 firms developed 457 programs. The great part of systems software are system support utilities. Software for general business management was the most of applications software. Most of these were user-friendly software for PCs or microcomputers and written in Cobol.

Table 27. Software developed by Local Software Houses (1986)

	Cases (%)
Systems software	83 (7.4)
System control	7 (0.6)
Communication control	15 (1.3)
Language compilers	12 (1.1)
System support	39 (3.5)
DBMS	10 (0.9)
Applications software	939 (92.6)
Business management	307 (27.5)
Marketing	218 (19.6)
Production management	135 (12.1)
Engineering	98 (8.8)
Public & Health	142 (12.7)
Office automation	39 (3.5)

Source: KIET/EIAK, 1986.

(b) Software Research in the National R&D Program

220. So far, local software houses developed very few advanced systems software such as operating systems and DBMS which requires high software technology. The government put great emphasis upon this area funding up to 100 per cent through the National R&D program since 1982. Software development projects which require advanced technology have been carried out by the Government alone (Government-led projects) or by the joint efforts of the Government and the private companies. Between 1982 and 1986, 84 projects were carried out with total expenditure of 16.2 billion Won of which the Government supported 55 projects fully with the total amount of 12.4 billion Won (see Table 28).

Table 28. Software Development Support by Korean Government Through the National R&D Program

Won million

	Gov't-led projects	partially supported projects
1982	337.5 (4)	276.3 (6)
1983	1,394.1 (6)	665.2 (4)
1984	2,547.0 (14)	649.3 (7)
1985	1,015.8 (11)	760.6 (9)
1986	7,080.0 (20)	1,489.0 (7)
Total	12,374.4 (55)	3,840.4 (29)

Notes: Numbers in ( ) are the number of projects.
Source: MOST, National R&D Program Yearbook, Korea, 1982-86.

(c) Assessment of the Technological Level

221. Systems Software: There is no single case of independently developed software by local firms reported in the area of systems software. The only development in the area is Korean OS, incorporating the capability of processing Korean into the imported OS such as MS-DOS, CP/M, XENIX, etc. However, Koreanization of UNIX is still on the way by ETRI (Electronics and Telecommunication Research Institute) and several other local computer manufacturers. A Korean model of computer hardware and accompanying OS are being on the process of development around ETRI. The technical level in these areas is going up continuously. However, it is still very behind in many areas such as automation and standardization of production process, system quality, testing technology, database technology, software management technology.

222. Applications Software: The bulk of applications software in use in Korea are for EDPS of general management such as pay-rolls, inventory management, accounting, etc. Around sixty per cent of local applications software demand is still supplied by in-house development. The local software development for science and technology is even lower. Most of software languages being used are Basic or Cobol, which shows a significant gap from up-to-dated technology such as the fourth generation languages and structured programming.

(3) Foreign Technology Import

223. Because of its high level of intelligence technology, the local software houses develop software through the technical license agreement with foreign firms. The recent rapid increase of foreign technology import and license agreement reflects the expansion of local software market and the increasing concern to this area.

224. The total of 45 cases of software imports were reported as of the end of 1986. (see Table 29) Software technology import started in 1970s, but has increased rapidly since the early 1980s. Because of no mandatory requirement for reporting software technology imports and furthermore complicated reporting procedure, the officially reported number of software technology imports is far from precision and tends to be highly underestimated. For example, there was no single case of import reported in 1984.

Table 29. Software Technology Imports by Korea

cases

	till 1981	1982	1983	1984	1985	1986	Total
Imports	9	2	4	0	8	22	45

Source: KIET/Ministry of Finance, Korea, 1987.

D. Legal Protection of Software

(1) Background

225. Since 1981, the legal protection of software in Korea has been requested by US in the Korea-US Commerce and Trade Conference. Furthermore, with the Commerce and Trade Tariff Reform of October 1984, the United States started to take retaliative actions, such as

import restriction or a retaliatory tariff, to countries which do not provide adequate legal protection of intelligence property.

226. This situation forced the Korean Government to take official actions for protecting the copyright of software. As a result, the Computer Program Protection Law of 37 articles has been passed to the Congress in December 1986, and the legislation came into force as of July 1, 1987.

(2) Contents of the Legal Protection

227. The Program Protection Law has been legislated in separation from the copyright law. But basically it acknowledges software as a kind of copyrights. The Article 3, on the foreign programs, extends the range of protection to the programs of foreign nationals with whose government Korea has established a formal diplomatic relation, with an exception that Korea may exclude a country in case its government does not provide the same protection.

228. Article 8, treating programs as a copyright, extends the period of protection up to 50 years. For full protection, however, all newly developed programs should be registered within one year from its development and at the same time be enlisted to MOST (Ministry of Science and Technology). This law also included articles on penalties upon infringement of copyrights.

E. Government Efforts for the Promotion of Software Industry

229. The information industry deserves a national promotion, because it is expected to become one of the key industries in 2000s, as well as recognized as a critical factor on overall efficiency of various social functioning, and thus for building welfare society. Information industry is linked horizontally to, and vertically with, a number of other industries so that development of information industry is critical to the productivity improvement in many other industries. Recognizing these aspects of the information industry, Korean Government pays due attention to implement a systematic and effective measures for the promotion of software industry, such as R&D, manpower training, expansion of computer usages, and establishment of administrative infrastructure.

(1) Software Technology Development

230. For the effective promotion of software industry, development of relevant software technology, such as automation technology for software production, and systems software technology, is required. The Government has taken the initiative of developing software technology as a part of the National R&D Program since 1982. Importation of foreign technology is also encouraged. In addition,

a large software development project, the SUPER (Software Usability and Productivity Enhancement Research) project, is being undertaken as a national project.

231. The SUPER project is a R&D program for enhancing the software usability and productivity, driven by the Government with wide participation of private companies, university research centers, and government research organizations. Major research areas of the Project include software engineering technology, systems software, artificial intelligence, high-level applications software technology including CAD/CAM, among others. The planned expenditure for the Project is 900 billion Won until 2001. In 1988, the first year of the project, the Government invests 3 billion Won.

(2) Software Promotion Law

232. To promote the software industry strategically, the Government has legislated a law. The Software Promotion Law, which came into effect as of July 1988, provides a basis for the Government to support the local software industry. It was legislated to expedite software development and to have it come into wide use. Followed are a few distinctive government actions taken or under considerations by the Law.

233. Operation of the Council of Software Industry Promotion is prescribed in the Article 4. The Council will be composed of government officials, scientists, and industrial experts for software development and productivity improvement, manpower development, and building up software development environment.

234. Based on Article 9, a functionally specialized software complex as an industrial base for software is going to be established in Seoul area. In the complex, the software companies are able to specialize in one area, and thus raise productivity, as more software and software-related companies are placed in the limited area. The complex eventually will be a platform for expansion to the international market.

235. In institutional aspects, for the effective promotion of the software industry, the Government is considering to guarantee loans from domestic commercial banks to software companies without any collateral (Article 10), and build up a system for implementation of software quality assurance (Article 7). Furthermore, a guideline for estimating the software development costs is prepared so that software is valued adequately (Article 8). This will restrict dumping, reduce the bankruptcy of software houses, lead to orderly distribution of software, and eventually lead to improvement of software quality.

236. To utilize efficiently the scarce local software development resources, and to assist marketing local software products

domestically and abroad, a software information center is going to be established for effective software information exchange. Without up-to-dated information on the software market and technology, domestic software houses cannot be competitive in the global market. Governments should provide the industry with the most recent information on market and technological changes. Korean government partially sponsors the software related trade associations such as Korean Software Industry Association and the Federation of Korean Information Industry to investigate such matters and distribute information.

(3) Standardization

237. The standardization in information industries is being undertaken in the direction of compatibility improvement among different hardware equipment and smooth interface of hardware and software, so as to increase the software usability. In 1987, the existing standard codes for Korean and Chinese languages were revised to maximize the information exchangeability in association with the national computer network systems under construction. Although, at present, Korea has set up 91 standards in information industry, it is not enough compared to the advanced countries. (see Table 30) Korean Government continues to build up new standards and revise the existing standards in accordance with the recent environmental changes. Recently, the Government decided to adopt the OSI (Open System Interconnection) standards for local computer networks.

Table 30. Standards in Information Industry

	ISO	US(ANSI)	Japan(JIS)	Korea(KS)
Number of standards	141	130	117	91

Source: MOST, Science and Technology Yearbook, 1988.

(4) Manpower Development

238. With the expansion of the social and industrial informatization, demand for professional personnel in this area is projected to increase from 10,000 in 1986 to 230,000 in 2001, among whom software scientists and engineers will be 31,000. The Government is planning to meet the expected demand through increasing enrollments in

related departments in universities.[20] Even though there are many private training schools specializing in computer education, the Government does not subsidize directly the private training schools for manpower development.

Table 31. Manpower Demand Forecast of Information Industry in Korea

persons

	1991	1996	2001
Scientists	1,600	3,300	5,600
Engineers	13,600	18,500	25,700
Total	15,200	21,800	31,300

Source: MOST, Science and Technology Yearbook, Korea, 1988.

(5) Establishment of Infrastructure

239. Proper understanding and social recognition on computers are the most important factors in wider dissemination of computers and building-up the age of information society. To establish the social infrastructure of information industry by 2001, a nation-wide information network, connecting even every household, is under plan. A new measure for computer safety and privacy protection is being developed to deal with the possible disfunction of wide dissemination of information.

240. Furthermore, promoting autonomous development activities within information industry, the Government supports software-related organizations, such as Korea Information Science Association, the Federation of Korean Information Industries, the Korean Software Industry Association, the Korea Software Research and Development Association, and the Electronics Industry Association of Korea.

---

[20]. In 1987, total of 140 colleges and universities in Korea open 337 Information Science related departments - Department of Electronic Computer Science, Department of Electronic Engineering, Department of Statistics, Department of Communications Engineering, Department of Electrical Engineering, etc. The departments have the total enrollment of 36,131 students. The enrollment in 132 electronic data processing related departments only is 14,695.

Chapter 6

PROMOTION OF SOFTWARE INDUSTRY IN DEVELOPING COUNTRIES

241. In adapting to the age of information society, each developing country must find its own path. Given the diverse backgrounds and conditions of the various developing countries, a variety of routes may be designed. In this chapter, the environmental factors common to the developing countries will be first discussed with respect to the promotion of the software industry, and discussions on the policy measures for them will follow, based on advantages and constraints intrinsic to the countries.

A. Environmental Factors In Development Of Software Industry

242. There are several ways in which a country develops software industry. Before directly getting into this issue, however, it is necessary to discuss the pros and cons for developing countries in terms of the factors which can affect the development of software.

(1) Internal Disadvantages

243. There are several internal factors that can hold back the development of a software industry. One of the difficulties stems from the narrow domestic market. Software market is strictly subject to the utilization of computer hardware equipment. Computer software is of no use without hardware. But hardware industry activity in the most of developing countries is very limited.

244. Another problem is the low level of software technology in developing countries. This has several aspects, one being the scarcity of competent programmers, due to the poor educational facilities in developing countries in comparison with advanced countries. Furthermore, it is difficult for well-trained specialists in these countries to get adequate experience to develop their capabilities and increase their expertise.

245. It is also necessary to build up industrial infrastructures which are directly relevant to the development software industry. These include standardization, telecommunication systems, and informatization of socioeconomic systems. Without all these conquered, it is very difficult to developing the industry, and to be competitive in the global software market. Therefore, support measures in the following will focus on these points.

(2) Internal Advantages

246. Nevertheless, developing countries do have some strong points in developing their own software industry. One asset is the affluent, relatively inexpensive labor which can be trained as software specialists. It is possible, in a developing country, to find cheap labor which can be well trained within a short time, thus reducing the production cost of software. Upward pressure on wages, combined with shortages of skilled manpower and the staggering productivity, creates good opportunities for firms in developing countries to perform contract programming and consulting services for firms in the developed countries.

247. Furthermore, development of applications software requires a full understanding of cultural aspects (mostly linguistic) of the user. This factor is a competitive advantage to local firms against foreign companies.

B. Strategical Approaches for Software Development

248. Promotional strategies for the software industry in developing countries may be different from country to country. There would be common strategies and policy options available to them, as well as native ones by country, due to the differences in industrial background and the environment of the country. In the following, common strategies and policy measures are discussed.

(1) Software Market Expansion

249. The fact that there is a large domestic software market, contributes to the development of software industry in the country. Conversely, the narrow domestic market is an obstacle to the development of the industry. Therefore, governments should have initiative to enlarge the local market at least in the initial phase of the promotion of software industry. In this respect, government procurement policy can greatly affect the expansion of the domestic market.

250. Dissemination of the hardware equipment is vital to the software market expansion through the construction of socio-economic information networks. This approach makes information technologies popular and widely used. In order to raise transferability, transportability, and usability of software which all together lead to the market expansion, government organizations must pay due attention to building-up standards of binary codes of their own languages and hardware/software interfaces. All of these endeavors combined will expand the range of software users.

251. The problem stemming from the narrow domestic market may be avoided by leaning to the overseas markets. In order to join foreign markets and sustain competitive strength in them, however, software must be produced at low cost and high quality. The former makes no problem, but the latter seems to be very difficult to get over. It requires skilled manpower and advances in information technologies. It is not possible to get it over without active R&D.

(2) Technology Development

252. Having access to advanced technologies and building-up technological capabilities, the developing countries may approach in three ways, that is, indigenous R&D, technology importation, joint ventures. Each of these will be discussed below, with emphasis on its own characteristics.

253. Indigenous research and development is an option which has to be pursued eventually in all the developing countries, if the situation permits. There is no objection in favor of local development of technology. A country should have its own technology base. Even when importing technology, it is required to assimilate it. Since most of the developing countries inevitably start with such a small scale of economy in the initial stage of software development, they hardly attract advanced countries to transfer technology. Furthermore, indigenous research activities are in some cases the only way to take such as development of technologies uniquely applicable to a certain region due to cultural and linguistic aspects.

254. Secondly, direct importation of software technology is regarded as a viable alternative by many developing countries whose technological base is negligible. If the importing party can negotiate front-end technology, a much higher take-off plane is insured. And it is possible to leap fast and easily to higher level technology, since information technologies relating to software development are rapidly changing. The developing countries can benefit considerably from hands-on training agreements in such deals.

255. The third way of technology acquisition is through joint ventures. One of the reasons for established industries to attempt joint ventures, is to make use of the marketing ability of the host party. Whether a partnership with a developing country with a weak technology base is attractive in the area of software is an open question. Many countries hold the view that there is nothing alien about technology, and what is most prudent for developing countries, is to adopt and absorb the latest available technology. Technology transfer through joint ventures is an effective approach open to developing countries when access to the frontier technology is limited by the policy of developed countries.

256. Each of the approaches in the above has its own pros and cons.

Therefore, developing countries may opt to mix the approaches. Whatever options are chosen by a developing country, the major task before them will be to achieve software technology in all the related sectors: telecommunications, control systems, transportation, manufacturing, defense industry, if these systems are intended to be anywhere near up-to-date and self reliant.

(3) Training of Technical Manpower

257. Trained human resources is a critical determinant to competitiveness in the software industry. As developing countries move to more sophisticated products, the demand for highly skilled manpower will grow accordingly. Several policies are conceivable to develop human resources in this area.

258. In the long term perspectives, there should be provided more opportunities for people to be trained in the national education system. Colleges and universities open more courses related to the computer science.

259. An approach available to government in the short-term perspectives is to encourage local software companies to train their staffs in fields such as information engineering, computer science, and management science, possibly by offering financial incentives to partially offset educational expenses. Special incentives can be offered to the firms supporting the education of their personnel in local university.

260. Governments may also operate vocational training centers specializing in computer software development, or may support private training schools. Through these organizations, low-level software manpower can be trained in a relatively short time, who can build up their skill on site training later.

(4) Development of Standards

261. The development of standards is of paramount importance in the promotion of Information Technology industries. Most areas of standardization are best left to international standards organizations, others may be best relegated to free-market de-facto standards. Governments have a role too, particularly with regards to access to the airways and monopoly telephone networks.

262. It is appropriate for governments in developing countries to mandate standards for the binary equivalents of their native language characters, and the code sequences that will switch a program from ASCII to the local translation; and the electronic standards for devices that are to be hooked into their national telephone systems. Regulation in this area should be sensitive to technological and market realities, or you may be faced with setting standards that

are obsolete and counter-productive. It is advisable to form an organization within the government authorities which are responsible for monitoring the changes in international standards, and setting and managing their own standards.

C. Policy Measures for Software Industry

263. In most cases, governments play a significant role in the development of an industry. First of all, in the United States, government support helped to establish the solid foundation of the industry and fostered its growth thereafter; the Japanese government also led to develop and improve the technological capability of the industry; government authorities had initiative to structure the industry in Western Europe, and promoted it in Asian developing countries, such as Korea, Singapore, and India.

264. There are various policy measures and programmes which have already been used in developed countries, in order to promote the software industry. Here, a few examples of incentive systems are introduced in terms of export, investment, R&D and manpower training.

(1) Preferential Treatment of Local Software Firms

265. Based on the discussions in the previous sections, the following measures are considerable: reduced income tax for software sales; accelerated depreciation allowance and investment tax credit for capital expenditures; exemption from tariffs for capital goods not locally produced; preferential loan facilities. All of the measures are aiming to help local software companies financially in the early stage of growth.

(2) Incentives for Export Promotion

266. The growth of the software industry in the long-run depends largely on the competitiveness in the global market. The export measures which all exporters automatically receive should be extended to the software industry. Some of these are listed: income tax reduction on income earned from direct or indirect exports; double deduction for export promotion expenditures from taxable income; export credit financing. Through the measures, local firms accumulate the ability to compete in the global software market.

(3) Promotion of Local R&D Activities

267. The buildup of local R&D capabilities is critical in the long-term growth of the software industry in developing countries. The government needs to serve as a catalyst for the development of an

indigenous R&D capacity. Moreover, it is advisable to encourage collaborative R&D, in view of the limited human, financial, and physical resources available.

268. The major incentives for the promotion of technology and R&D activities are: double deduction for R&D expenditures ; investment tax credits on R&D investment; encouragement of cooperation between industry and universities; inducement, through administrative guidance, of minimum R&D expenditures by large firms in terms of a percentage of the total sales revenue.

D. Concluding Remarks

269. It is obvious through discussions that the software industry is worthwhile to promote in a country in view of the enormous growth potential of the industry, as well as its impact on other industries. When a country promotes the industry, it is utmost important to consider the balanced development of both demand and supply of software.

270. In the demand side of software, appropriate policy measures for more utilization of computer systems, government procurement, standardization, and exports should be developed. In order to increase the software supply capability of local companies, governments are encouraged to support R&D and manpower development activities. Needless to say, some policy measures to support local software houses will be effective.

271. Developing strategies for the promotion of software, the approaches taken by Korea as well as advanced countries such as Japan and the United States will be of great help. However, above all the most important is a complete understanding of software industry itself, and internal advantages and disadvantages in the development of software industry. Whatever policy options may be developed, they should be based on the environmental factors, strength and weakness, which a nation faces.