

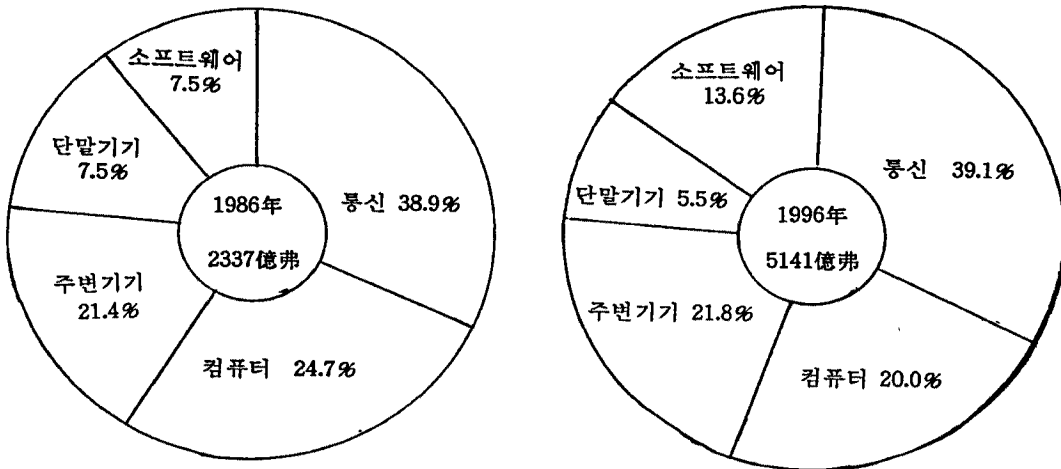
第 2 節 소프트웨어 産業

1. 世界 소프트웨어 産業動向

컴퓨터가 商用化된 1950年代 以後 컴퓨터를 中心으로 한 情報産業의 規模는 1986年 現在 約 2,337億弗 以上으로 크게 成長하고 있다. 또한 産業의 底邊擴大에 따라 이미 컴퓨터産業이라는 單一分野를 넘어, 1996년에는 5,141億弗이 넘을 것으로 豫見되어 情報處理産業(Data Processing Industry)이라는 새로운 巨大産業을 만들어 낼 것으로 보인다.

<圖表 II - 2 - 101>

世界情報處理産業의 市場動向 및 豫測



資料 : ADL '88

<圖表 II - 2 - 101>은 多岐化가 進行되어은 世界情報處理産業의 各 部門別 成長豫測을 나타내고 있다. 1986년부터 1996년까지의 10年間 産業規模는 2.2배에 가까운 成長이 있을 것으로 豫測된다. 소프트웨어部門의 比重은 持續적으로 增加할 것으로 展望된다. 컴퓨터部門은 그 比重이 24.7%에서 20.0%로 減少하나 이 産業에 있어서 傳統的인 先導的 入場을 維持할 것으로 보이며 소프트웨어는 가장 높은 成長을 보일 것으로 전망된다.

ADL의 資料에 의하면 世界 소프트웨어市場은 1986년에는 174億弗에 불과했으나 1991년에는 363億弗로 늘어나 年平均 15.8%의 매우 急速한 成長을 이룩할 것으로 展望하고 있으며 '96년까지 11年동안 14.9%의 높은 成長率을 보인 것으로 보인다. 이는 소프트웨어가 컴퓨터 하드웨어나 他産業에 비해 월등히 높은 成長을 할 것이라는 점을 示唆하는 것이며 未來指向的인 面에서 그만큼 重要한 分野임을 뜻한다.

<圖表 II - 2 - 102>

世界 情報産業市場

(單位 : 億 \$, %)

區 分	'86	'91	'96	年平均 成長率
Computer	577	770	1,028	5.9
端 末 機 器	175	238	285	5.0
周 邊 機 器	501	740	1,120	8.4
소 프 트 웨 어	174	363	700	14.9
小 計	1,427	2,111	3,133	8.2
通 信 機 器	910	1,360	2,008	8.2
計	2,337	3,471	5,141	8.2

2. 主要國의 소프트웨어 産業動向

가. 美國

1986年 美國의 소프트웨어 産業은 540億弗(前年非 16% 增加)에 달하며 향후 이러한 정도의 成長은 계속될 것으로 보인다.

(1) 企業數

소프트웨어産業關聯 企業數는 1986年 7,532社이며 전년비 3% 增加하였다. 이를 企業形態別로 보면 Processing Service 2,110社, 소프트웨어 프로덕트는 2,705社, Professional Service는 1,555社, Turn-Key System은 1,162社이다. 기업의 경향을 보면 소프트웨어 프로덕트 分野의 增加와 턴키시스템 分野의 減少가 현저하였고, Processing Service는 점차 減少, Professional Service는 집중하고 있다.

<圖表 II - 2 - 201>

美國의 S/W企業數 推移

區 分	'84	'85	'86	增 加 率
Processing Service	2,150	2,121	2,110	△0.9
Software Product	2,500	2,488	2,705	4.0
Professional Service	1,450	1,475	1,555	3.6
Turn Key System	1,300	1,229	1,162	△5.5
計	7,400	7,313	7,532	0.9

資料 : INPUT

(2) 賣上高

賣上高는 Processing Service가 가장 많은 213億 \$ 이나 成長率은 13%로 全體平均 伸張率인

16%보다 낮다. 다음으로는 소프트웨어 Product의 148億 \$로 成長率은 24%로 가장높은 伸張을 보였다. Professional Service도 成長率 20%를 보여 111億 \$에 이른다. 턴키 시스템은 69億 \$로 전년대비 8% 成長하여 가장 낮은 增加率을 보였다.

<圖表 II - 2 - 202> 美國의 S/W賣出高 推移

(單位：百萬 \$, %)

區 分	'85	'86	成長率('85~'86)
Processing service	18,867	21,316	13
Software product	11,944	1,478	24
Professional service	9,240	11,084	20
Turn key system	6,267	6,853	8
計	46,418	54,037	16

資料：Input

(3) 利用 現況과 展望

使用者側에서 본 각 서비스 形態의 '86年 支出實績은 우선 Processing Service에의 支出이 가장 많은 183億 \$로 全體의 33%로 접하고 있다. 다음으로 Software Product의 162억1\$, Professional Service 123億 \$, Turnkey System 78億 \$이다.

1992년에는 software Product가 Processing Service를 上廻하는 522億 \$로 38%를 접한 것으로 보이며 다음으로 Processing Service 382億 \$, Professional Service이 339億 \$, Turnkey System 133億 \$로 豫測된다.

<圖表 II - 2 - 203> 美國의 部門別 S/W市場 動向

區 分	'86	'87	'92	年平均 增加率 '92/'86
Processing service	18.3	20.9	38.2	13.0
Software Product	16.2	19.9	52.2	21.5
Professional service	12.3	14.7	33.9	18.4
Turnkey system	7.8	8.6	13.3	9.3
計	54.6	64.1	137.6	16.7

聯邦政府和 民間에 있어서 情報서비스 分野 支出은 '86年 民間 498億 \$ 연방정부 48억불로 91 : 9水準이다. 또한 '87년에는 各各 587億, 53億 \$로 92 : 8, '92년에는 1281億 \$, 96億 \$로 93 : 7이 되는 등 점차 民間部門 需要가 增大되어 나갈것으로 보인다.

나. 日 本

日本の 情報處理産業은 '87년부터 向後 3年間 毎年 14% 前後의 伸張을 나타낼 것으로 豫想된다. 소프트웨어開發販賣業을 包含한 情報處理業體가 1986년에는 2,808業體로 調査되었는데 이는 10年前보다 約2倍이상 増加한 것이며 從業員數역시 19萬3千餘名으로 10年間 3.3倍 늘어났다. 情報서비스 販賣額은 '86년에 1兆9,159億圓 規模에 달해 10年 前보다 6.2倍 증가한 높은 伸張率을 나타냈다. 이에 따라 從業員 1人當 年間 販賣額은 965萬圓에 이르러 10年間 2倍 수준으로 擴大되는 成長實績을 記錄하였다.

1986年中 情報 서비스類別 販賣額을 보면 소프트웨어開發 및 프로그램 作成이 9,127億圓으로 47.6%의 가장 큰 比重을 보였고 受託計算이 4,278億圓, 시스템 管理運營受託이 1,443億圓으로 各各 22.3%, 7.5%의 比重을 나타냈다. 한편 11年前 對比로 보면 역시 소프트웨어開發 및 프로그램 作成을 통한 販賣額이 21.7倍의 가장 높은 伸張率을 記錄하였고 情報提供서비스販賣額도 8倍의 增加幅을 維持 한 反面, 머신타입 販賣는 12%가 감소하였다.

<圖表 II -2-204> 日本情報서비스業 業體數와 年間 販賣額

年 度	調 查 對 象 體 業	從 業 員 數 (名)	年 間 販 賣 額 (百 萬 圓)	從業員 1人當 年間 販賣額 (萬圓/名)(指數)
1981	1,801	105,898	805,692	761
1982	1,864	113,414	911,907	804
1983	2,148	127,978	1,095,301	856
1984	2,549	153,474	1,385,974	903
1985	2,556	162,010	1,561,829	964
1986	2,808	193,522	1,915,900	965

資料：日本 通產省

<圖表 II -2-205> 日本 情報서비스類別 賣出額 推移 (單位：百萬圓)

區 分	1975年	1980年	1984年	1985年	1986年
計	275,091	669,844	1,385,974	1,561,829	1,915,936
委 託 計 算	103,565	204,343	377,790	390,713	427,826
S/W開發·프로그램 作成	42,082	153,985	512,398	658,030	912,747
키 편 치	41,214	74,205	108,116	109,650	120,324
머 신타입 販賣	13,790	15,345	17,467	19,164	12,073
시스템等管理運營受託	24,457	104,103	127,427	117,699	144,323
情報 提供 서비스	14,376	44,059	96,654	100,762	114,306
各 種 調 査	22,237	38,676	63,117	56,980	72,989
其 他	13,371	35,128	83,003	108,831	111,352

資料：日本 通產省

1986年中 日本 情報서비스業界의 賣出을 需要處別로 區分하면 鑛業, 製造業에 대한 販賣額이 5,693億円, 金融·運輸業에의 實績이 4,307億円 水準으로 各各 37.6%와 28.4%의 높은 比重을 占有하였고 個人(一般 消費者)에 대한 販賣額은 48億円 規模로 0.3% 比重에 그쳤다. 86年 實績을 11年前과 比較하면 支社나 同種業者에 대한 賣出額이 11.7배와 19.6배씩 크게 늘어 왔으며 서비스業과 製造業에 대한 販賣額도 5.8배와 9.1배씩 伸張되어 왔으나, 公共部門에 대한 賣出額 伸張은 3.7배의 低調한 比率을 나타내고 있다.

職種別 從業員數를 보면 프로그래머가 60,266名으로 30.4%, 시스템 엔지니어가 48,095名으로 24.2%의 比重을 차지하였는데, 11年前 規模와 比較하면 시스템 엔지니어가 7.3배, 프로그래머가 5.5배로 가장 큰 幅의 增加率을 示現하였다. (<圖表Ⅱ-2-207>)1986年을 基準으로 하면 日本에서의 소프트웨어開發 프로그램 賣出은 過去 11年間 21.7배 增加한 反面, 시스템엔지니어와 프로그래머는 6.2배 增加하여 시스템엔지니어와 프로그래머 1人當 소프트웨어 賣出額은 1975年 242萬円에서 1986年 842萬円으로 3.5배 伸張되어 生産性이 크게 提高되고 있는 것으로 나타났다

日本 情報서비스業 從業員數는 1980年에 9萬3千餘名, 1986年에 19萬8千餘名으로 增加한데

<圖表 I -2-206>

日本 情報서비스業 需要處別 賣出額

(單位：百萬円)

區 分	1975年	1980年	1984年	1985年	1986年
計	275,091	669,844	1,385,974	1,561,829	1,914,939
農 林 · 水 山 業	4,510	6,989	20,858	17,567	14,176
鑛 業 · 製 造 業	62,282	173,614	366,443	429,123	569,306
都 · 小 賣 業	44,074	81,408	148,709	172,296	171,890
建 設 · 不 動 產 業	6,694	11,386	21,769	26,205	34,260
金 融 · 保 險 · 運 輸 電 氣 · 가 스 · 水 道 業	55,872	120,766	288,540	304,472	430,698
서 비 스 業	20,978	63,521	127,068	157,159	121,922
公 務	47,706	96,220	158,481	157,969	174,366
個 人 (一 般 消 費 者)	669	806	2,645	2,868	4,849
其 他	15,687	34,055	63,140	85,664	108,829
同 業 者	11,499	53,322	104,700	118,714	225,428
本 社 · 支 社	5,119	27,748	83,621	90,791	60,217

資料：日本 通產省

<圖表 II - 2 - 207>

日本 情報서비스業 職種別 從業員數

(單位: 名)

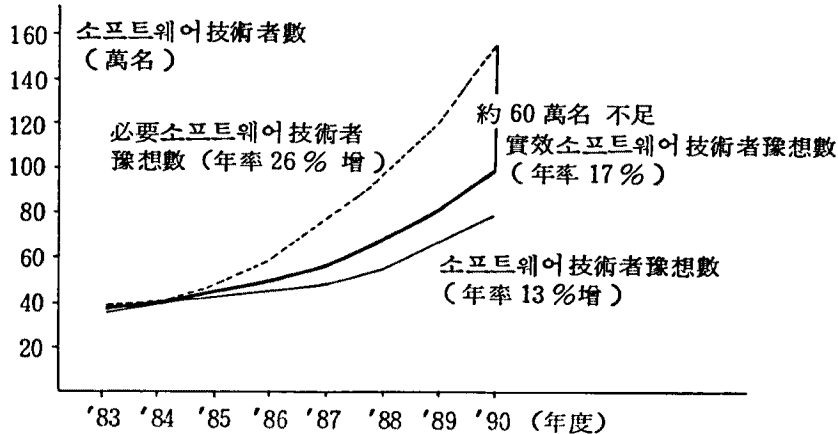
區 分		1975年	1980年	1984年	1985年	1986年
專 業 所 數(개사)		1,276	1,731	2,549	2,556	2,808
政 務 報 務 部 門 서비스	計	57,164	93,271	153,474	161,010	198,522
	個人事業主·家族 從業員 또는 有給職	2,893	4,539	5,172	5,244	4,408
當 時 從 業 員		54,271	88,732	148,302	156,766	192,330
職 種 別	計	57,164	93,271	153,474	162,010	198,522
	管 理 部 門	7,133	11,955	17,265	17,780	23,045
	研 究 員	1,844	2,632	3,003	2,806	3,356
	시 스템 엔지니어	6,572	13,673	29,233	32,978	48,095
	프 로 그 래 머	10,849	19,968	43,745	48,544	60,266
	오 퍼 레 이 터	7,394	11,298	15,745	16,196	18,096
	키 편 치	15,772	22,300	26,300	25,441	26,450
	其 他	7,600	11,445	18,183	18,256	19,124

資料: 日本 通產省

이어 1990년에는 25萬6千餘名으로 '80年 對比 2.8倍 늘어날 展望이며 시스템엔지니어와 프로그래머는 1990년에 13萬7千餘名으로 '80年보다 4倍가 增員될 것으로 豫側되나 人力需要에는 크게 不足하리라는 展望이다 (<圖表 II - 2 - 208>).

<圖表 II - 2 - 208>

日本の 소프트웨어 人力需要갭 豫側



資料: 日本 通產省, 「特定서비스産業實態調査」.

<圖表 II - 2 - 209> 日本의 業態別 S/W 賣上高 推移 (單位: 百萬円, %)

區 分	1 9 8 0	1 9 8 5	1 9 8 6	年平均增加率
計	669,844	1,561,829	1,915,939	19.1
소프트웨어業	141,561	660,473	875,519	35.5
情報處理서비스業	432,568	691,998	823,049	11.5
情報提供서비스業	44,895	97,993	107,536	15.7
其他情報서비스業	50,821	111,365	109,835	13.7

資料: 日本 通産省

다 유 럽

유럽의 소프트웨어市場은 1986年 235億 \$ 이며 IBM을 필두로하는 美國系 企業에 의해 占有되고 있다. 小型機分野, 個人用컴퓨터分野에서는 유럽企業과 競争을 벌이고 있다. 그러나 소프트웨어開發, 提供 및 受託計算 서비스등의 情報處理서비스市場으로는 유럽企業이 꽤 活躍을 보이고 있다. IDC에 의하면 유럽 컴퓨터소프트웨어 서비스市場은 1985年 197億730萬 \$ 에서 '86年 235億5,350萬 \$ 로 年間 19%씩 伸張해왔다. 이 成長率은 1987年에도 변하지 않는다고 ECSA는 보고 있다. 國別로 보면 西獨이 46億2100萬 \$ 로 유럽全體의 20%(前年比 19%), 프랑스 50億8,400萬 \$ 로 유럽全體의 22%(前年比 22%增加), 英國 38億1,700萬 \$ 로 16%(前年比 16%), 이태리 24億9,300萬 \$ 로 全體의 12%(前年比 10%)를 점유하였으며 서비스 形態別로 보면 패키지 소프트웨어는 83億9,400萬 \$ 로 傳年比 28%增加, Custom soft/consultant는 67億8,300萬 \$ 로 20%增加, Training은 10億8,600萬 \$ 로 22%增加, processing service로 70億2,100萬 \$ 는 9%증가, F.M은 2億5,100萬 \$ 로 16%增加해 全體의 19% 增加할 것으로 보인다. Processing service가 西유럽매상고의 約 30%를 점하고 9%의 成長, 이중에도 특히 Local Batch Service는 1985年 18億 8,300萬 \$, '86年18億8,600萬 \$ 로 現狀維持 하였다.

<圖表 II - 2 - 210> 유럽의 國別 S/W市場 (單位: 百萬弗, %)

區 分	1 9 8 5	1 9 8 6	增 加 率
西 獨	3,883	4,621	19
프 랑 스	4,167	5,084	22
英 國	3,291	3,817	16
이 태 리	2,266	2,493	10
其 他	6,100	7,539	23.6
計	19,707	23,554	19.5

<圖表 II-2-211>

유럽의 形態別 S/W市場

(單位: 百萬弗,%)

區 分	1985	1986	增 加 率
패 케 이 지 서 비 스	6,558	8,394	28
Custom soft, Consultant	5,653	6,783	20
Tranning	890	1,086	22
Processing service	6,441	7,021	9
其 他	165	270	64.6
計	19,707	23,554	19.5

라. 臺 灣

(1) 概 要

臺灣의 소프트웨어産業의 歷史는 約10年 程度이다. 1986年 賣出額은 93百萬\$이며 從業員數는 4,000名으로 1人當 賣出額은 1萬5千\$이다. 또한 投資額은 6千萬\$이며 '86年 輸出額은 7百萬\$이다.

<圖表 II-2-212>

臺灣의 S/W市場

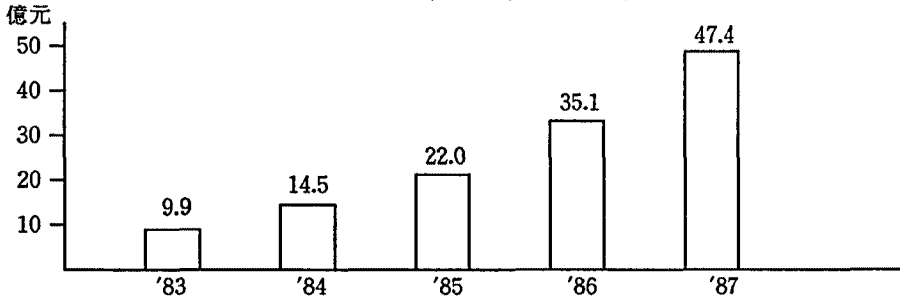
區 分	1986	1987	增 加 率(%)
生 產 額(백 만 \$)	93	126	35.5
業 體 數 (개 사)	250	270	8
人 力 (名)	4,000	5,300	32.5
1人當生産額(千\$)	23.3	23.8	2.1
投 資 額(백 만 \$)	60	75	25
輸 出 額(백 만 \$)	7	16	128.6

資料: MIC

(2) 賣出推移

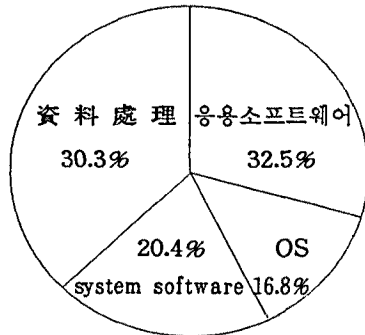
臺灣의 '87年 賣出額은 約47.4億元이며 4年前의 9.9億원에 비해 9배가 넘어 年平均 47.9%의 成長을 보이고 있다.

<圖表 II-2-213> 臺灣의 S/W賣出額 推移



이러한 소프트웨어를 分野別로 보면 응용소프트웨어가 全體의 32.5%, OS가 16.8%, System software가 20.4%, 資料處理 30.3%를 차지하고 있다.

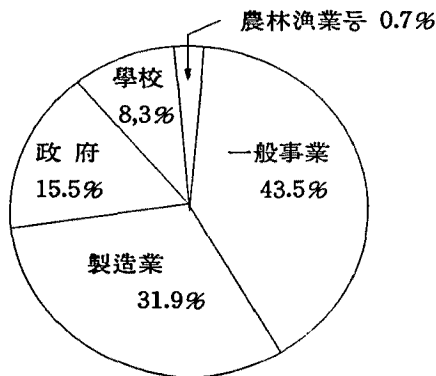
<圖表 II-2-214> 臺灣의 S/W分野別 賣出額 構成



(3) 市場 構造

臺灣의 最大 소프트웨어市場은 製造 및 一般事業으로로 全體의 75%를 차지하고 있다. 그 다음으로 政府需要가 15.5%, 學校가 8.3%, 農·林·漁業 등 其他가 0.7%를 차지하고 있다.

<圖表 II-2-215> 臺灣의 S/W市場 構造

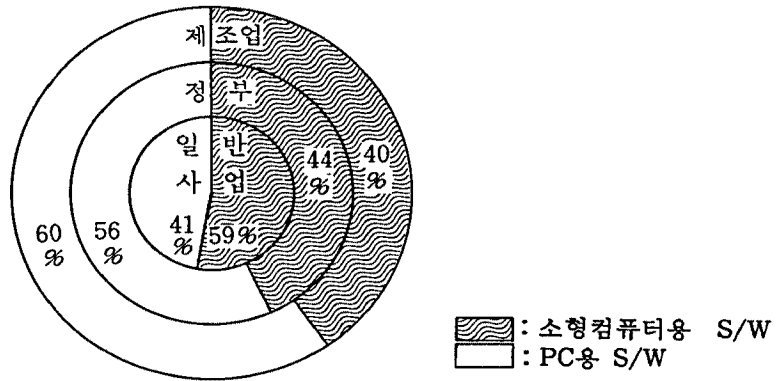


資料 : MIC

이러한 市場을 使用컴퓨터別로 보면 PC用 소프트웨어가 全體의 40~60%를 차지하고 있으며 나머지는 Micro以上の 컴퓨터用 소프트웨어이다.

<圖表 II -2-216>

臺灣의 컴퓨터機種別 S/W市場



3. 最近의 技術開發 動向

가. 部門別 技術 動向

2000년대 高度 情報社會에서는 컴퓨터, 소프트웨어, 通信 등의 情報通信 技術이 基盤構造를 形成하게 되며, 現在의 컴퓨터보다 高速, 高成能일 뿐만 아니라 使用이 簡便한 컴퓨터가 低廉한 價格으로 大量 普及되고, 소프트웨어 패키지 普及 擴散으로 一般인도 쉽게 소프트웨어를 利用할 수 있게 될 것으로 豫想된다. 近來에 들어 소프트웨어 技術은 여러 分野에 걸쳐 커다란 發展을 이룩하고 있는 데, 시스템 소프트웨어 部門에서는 컴퓨터의 處理 速度와 異機種간의 互換性を 높이기 위한 努力이 進行되고 있고, 應用 소프트웨어 쪽에서는 單一 業務處理 水準을 벗어나 綜合的인 經營情報處理를 指向하는 趨勢이며, 專門家 시스템이나 自然語 處理등과 같은 人工智能 소프트웨어 技術도 많은 進展을 보이고 있다. 또한 소프트웨어 生産技術 側面에서는 開發環境 助성과 아울러 再使用, 自動化 프로그래밍 등과 같은 새로운 方法이 繼續 發表되고 있고, 소프트웨어 品質保證 技術도 活潑하게 研究되고 있다. <圖表 II -2-301>은 이제까지의 소프트웨어 技術의 發展 趨勢를 보인 것이다.

最近의 소프트웨어 技術의 一般的인 動向을 高級化, 高速化, 簡便化, 安定化, 自動化, 標準化의 側面에서 살펴보면 다음과 같다.

첫째, 컴퓨터가 文字 뿐만 아니라 圖形, 畫像, 音聲등과 같은 多樣한 情報를 認識하여 處理할 수 있게하며, 人間의 思考 能力과 비슷한 推論 技能을 갖게하는 등의 技術의 高級化가 推進되고 있다.

<圖表 II -2-301>

소프트웨어 技術의 變遷

年 度	소프트웨어 技術 內 歷
1957年代	機械語 프로그램 初歩的 引出力制御시스템
1958~1967	高級 言語 프로그램 一括處理 시스템 注文型 소프트웨어
1968~1974	多重 使用者시스템 데이터베이스 構造化 프로그래밍 소프트웨어 製品化
1975~1980	소프트웨어 工學 導入 工學的 開發技法 確立
1980~現在	第 4世代 言語 普及 竝列處理 시스템 네트워크 및 墳散處理 技術 經營情報處理 시스템 그래픽 및 CAD/CAM 技術 專門家 시스템 開發 소프트웨어 開發環境 構築 品質保證 技術
未 來	人工知能 시스템 소프트웨어 自動生産 ISDN 實現

둘째, 大量의 情報을 高速으로 處理하기 위하여 多重處理 및 竝列處理 소프트웨어技術과 通信網을 利用한 分散處理, 遠隔地 온라인 處理와 같은 高速處理 技術이 重要해지고 있다.

셋째, 複雑한 技能의 소프트웨어를 一般 使用者가 쉽게 利用할 수 있도록 하기 위하여 多樣한 使用者 인터페이스가 開發되고 있다.

네째, 社會的으로 소프트웨어가 차지하는 比重이 커짐에 따라 시스템 障礙나 데이터 損失로부터 使用者를 保護하고, 소프트웨어가 正確하고 安定的인 技能을 遂行할 수 있도록 하기 위한 努力이 進行되고 있다.

다섯째, 소프트웨어 開發의 生産性を 높이기 위해 多様な 開發支援道具들이 發表되고 있으며, 나아가 소프트웨어의 自動生産 技術에 대한 研究도 活潑히 進行되고 있다.

여섯째, 異機種 컴퓨터간의 互換성과 네트워크의 連動성을 確保하기 위해 運營體制나 네트워크의 프로토콜, 데이터 表現을 標準化하고, 소프트웨어 開發過程에서도 生産性, 品質 向上을 위해 여러 產出物에 대한 規格化가 進行되고 있다.

한편 소프트웨어 技術을 크게 프로그래밍 言語 開發, 시스템 소프트웨어 技術, 應用 소프트웨어 技術, 人工知能 소프트웨어 技術, 소프트웨어 開發技術, 品質保證 技術등으로 區分할 수 있는데 이들 各各에 대한 技術動向을 紹介한다.

(1) 프로그래밍 言語

프로그래밍 言語는 特定한 業務를 處理하기 위하여 使用者의 意思를 컴퓨터에게 傳達하고 그의 實行을 制御하는 手段으로 使用되어 왔다. 즉, 프로그래밍 言語는 소프트웨어를 作成하는 가장 基本的 道具인 셈이며 이에 따라 소프트웨어 技術과 프로그래밍 言語는 매우 密接한 關係에 있다.

初期 第 1世代의 機械語 및 第 2世代의 어셈블리 言語는 컴퓨터가 解讀하기는 쉬웠으나 프로그래머가 프로그램을 作成하기에는 아주 不便하였다. 점차로 프로그래밍 言語 技術과 컴파일러를 構成하는 技術이 發展함에 따라 프로그래머의 수고를 덜기 위하여 FORTRAN, COBOL, PL/1, ALGOL등과 같은 第 3世代의 高級 프로그래밍 言語를 使用하게 되었다. 그러나 이러한 言語들은 소프트웨어 工學의 原理를 充實히 反映하지 못해 소프트웨어의 信賴性, 移植性, 維持補修性등에 미흡한 점이 많았다. 이에 따라 소프트웨어의 設計 및 프로그래밍을 보다 效果的으로 할 수 있도록 소프트웨어 開發過程의 抽象化(abstraction) 技法이나 모듈화 原則을 充實히 提供하는 言語로서 C++, Smalltalk, Eiffel 등과 같은 個體指向 言語(object oriented language)가 나오고 있다. 이들 言語에서는 각각의 프로그램 모듈에서 데이터와 프로그램이 한 덩어리가 되어 하나의 目的 또는 對상을 構成함으로써 소프트웨어 開發過程을 體系化하여 維持補修가 容易하고, 使用이 便利하며, 프로그램 모듈을 標準化된 部品처럼 再使用할 수도 있게 되었다. 그리고 Ada는 美國防省에 의해 開發된 프로그래밍 言語로 大規模의 소프트웨어 開發에 便利하도록 소프트웨어 工學의 諸般 原理를 包括하고 있는 言語이다. Ada는 嚴格한 데이터 型을 提供하여 프로그램의 信賴性を 높이고, 패키지 概念을 통해 抽象化 技法이나 情報隱匿을 效果的으로 提供하면서도 프로그램의 效率性を 높이기 위한 特性을 고루 갖추고 있다.

한편 1980年代에 들어서는 FOCUS, NOMAD, RAMIS, UFO, MANTIS 등의 第 4世代의 非節次性 言語로 第 5世代의 知識基盤言語(knowledge based language)등의 새로운 言語들이 紹介되었다.

第 4세대 言語의 特徵으로는 生産性이 높고, 使用하기에 便利하며, 適用성과 效率성이 높을 뿐만 아니라 각종 컴퓨터 시스템간에 쉽게 移植할 수 있다는 점을 들 수 있다. 이러한 第 4세대의 不足 등이 言語는 既存의 3세대 言語의 文章構造에다 컴퓨터 그래픽의 向上된 對話式 스크린 技術, 強力한 데이터베이스 그리고 業務에 대한 諸般 規則을 廣範圍하게 갖추고 있는 辭典 등의 機能을 補完하여 開發되고 있다. 그러나 第 4세대 言語의 경우 標準言語의 不足과 데이터의 補完性, 書類化의 어려움, 데이터의 送受信 問題, 데이터의 貯藏容量問題, 패키지化 및 4세대 言語專門家の 不足이 解決되어야 할 先決課題로 대두되고 있는 實情이다. 第 4세대 言語는 프로그램 作成 速度가 5~10배 빠르다는 長點이 있어 現在의 趨勢로 나가면 1990년에 가서는 全體市場의 80%를 占有할 것으로 豫測된다.

第 5세대 言語는 第 4세대 言語의 開發 및 普及에 힘입어 自然語 프로그래밍을 目標로 發展하고 있다. 특히 最近의 人工智能 分野 중에서 知識基盤 시스템, EXPERT시스템, 自然語 處理 등에 效果的으로 使用할 수 있는 Prolog, Planner, Connviver 등과 같은 言語들이 普及되면서 이른바 第 5세대 言語에 대한 開發이 活潑히 推進되고 있다. 第 5세대 言語의 特徵은 Back tracking 이라 불리는 制御構造가 言語에 自動적으로 提供되고 있다는 점이다. 이러한 技能은 주로 주어진 問題를 解決하는 프로그램의 摸索過程에 必要한 技術로서, 일정한 時點에 特定 規則이나 條件을 選擇할 때 추후의 計算 結果가 원하는 結果가 아닌 경우 이전으로 돌아가 다른 規則을 設定할 수 있도록 하는 것이다. 그러나 이러한 機能만으로 完全한 言語를 構成할 수 없으며, 이외에도 竝列處理와 같은 多様な 制御構造와 데이터 形態들을 具備하여야 하는데 대다수의 第 5세대 言語들은 이러한 追加 機能을 완벽하게 갖추지는 못하고 있다.

(2) 시스템 소프트웨어 技術

시스템 소프트웨어 하드웨어와 密接히 連結되어 있는 基本 프로그램을 말하며, 시스템 全體의 動作狀態와 프로그램 實行過程을 監督하고 支援하는 運營體制, 컴퓨터간의 通信을 支援하는 通信制御 프로그램, 그외에 使用者들이 컴퓨터를 便利하게 使用할 수 있도록 도와주는 유틸리티 프로그램 등으로 構成된다.

初期의 컴퓨터 프로그램은 運營體制 없이 機械語로 作成되었으므로 프로그램 作成이 어려울 뿐만 아니라 많은 時間이 所要되는 短點이 있었다. 이에 따라 컴퓨터 産業을 主導해 온 IBM社에서는 人出力의 同時 操作과 自動制御 및 각종 프로그램 言語를 使用할 수 있는 IBSYS/IBJOB 등의 制御 프로그램을 開發하였으며, 1964년에는 여러 프로그램을 同時に 遂行할 수 있는 IBM 360 OS를 發表하였고, 1972년에는 假想記憶裝置를 實現한 SVS를 發表하였으며, 이어서 MVS를 發表함으로써 大型 컴퓨터의 運營體制는 IBM社에 의해 標準化 되어가고 있다.

이제까지 學校 및 研究所에서는 原始 프로그램이 公開되어 있는 UNIX 運營體制를 많이 使用해 왔는데, 情報 및 通信 分野의 市場調査 會社인 美國 양키그룹의 最近 報告는 UNIX의 市場占有率이 점점 擴大되고 있음을 보이고 있다. 個人用 컴퓨터에서 부터 大型의 汎用 컴퓨터에 이르기 까지 폭넓게 利用할 수 있는 점이 認定되어, 1984년에는 300個에 불과하던 應用 소프트웨어가 現在는 1,600個의 패키지 소프트웨어를 비롯하여 約 3,000個의 製品과 서비스가 提供되고 있고, 그 範圍도 科學, 엔지니어링 分野에서 事務 分野나 製造業 分野로 擴散되고 있다. 現在 UNIX 運營體制는 하드웨어 規模나 機關 및 團體에 따라 여러 種類가 있으나, 앞으로는 開發者인 AT&T와 規格統一을 指向하는 製造業體 그룹인 X/Open 등의 標準化 努力에 IBM도 協力할 것으로 豫測되어 '90年代 前半에 UNIX 運營體制가 모두 標準化되면 UNIX의 市場은 더욱 擴大될 것으로 豫想된다.

한편 각각의 運營體制에 대한 研究와 아울러 서비스가 高度化, 多樣化 됨에 따라 소프트웨어의 流通을 重視하여 運營體制에 대한 共通化, 標準化 努力도 活潑해지고 있다. 이제까지 運營體制의 種類는 수도 없이 많았고, 각각이 特定 分野에서 뛰어난 性能을 갖고 있었으며, 앞으로는 네트워크의 活性化와 함께 運營體制간의 連動性 確保가 切實해짐에 따라 運營體制간의 標準化 努力이 進行되고 있다. 大型 機種에서는 그 패밀리 안에서의 標準化를 우선 試圖하고 있으며, 異機種간에도 相互接續을 위한 標準化와 아울러 인터페이스 構成에 대한 努力이 進行되고 있다. 이를 위해 각 運營體制 위에 양과겹질 같은 새로운 소프트웨어 層을 입혀 모든 運營體制가 共通的인 對話의 通路로 使用하는 方法이 研究되고 있다. 이미 IBM PC와 그 互換機種의 프로그램을 UNIX가 돌아가는 工學用 워크스테이션에서 遂行할 수 있게 해주는 XDOS 소프트웨어가 使用되고 있고, 歐美에서 推進하고 있는 POSIX에서는 原始 프로그램 次元에서 應用 프로그램의 互換性을 支援하기 위하여 UNIX에 입각하여 標準 運營體制 인터페이스와 環境을 提供하고 있으며, MOSI에서는 應用 프로그램에 假想머신(Virtual Machine) 인터페이스를 提供함으로써 異機種 컴퓨터간에 프로그램의 互換性을 갖게하고 있다.

데이터 通信技術의 進歩에 힘입어 컴퓨터들이 온라인 네트워크化 되어감에 따라 通信制御 소프트웨어에 대한 技術開發도 매우 活潑해지고 있다. 컴퓨터 네트워크 管理는 勿論 多樣한 最新의 서비스를 提供하기 위한 소프트웨어가 많이 開發되고 있으며, 컴퓨터 네트워크의 互換性과 連動性을 確保하기 위하여 OSI 모델을 基準으로 通信프로토콜의 標準化가 進行되고 있다. 또한 각종 데이터를 最適 常態로 分散하여 制御하면서 필요에 따라 네트워크를 통해 檢索하고 統合할 수 있는 分散處理 技術도 크게 發展하고 있다. 앞으로의 情報化 社會를 위하여 文字情報 이외의 圖形, 畫像, 音聲情報 등을 자유롭게 電送하고 處理할 수 있는 멀티미디어 技術과 分散 데이터베이스 技術, 相互運用 네트워크 시스템 技術을 開發하고 이를 基礎로 多樣한 情報器機 및 시스템의 相互運用을 可能케 하는 綜合情報通信網(ISDN)을 構築하려는 努力도 꾸준

히 進行되고 있다.

最近에는 HP, DEC, 아폴로, IBM 등 美國會社와 서독의 지멘스, 닉스도르프, 불란서의 Bull Group 과 공동으로 '88年 5月 17日 7個社의 出資로 소프트웨어 開發會社를 設立, 發表하였다. 이 會社는 Oren Software Foundation(OSF)로 事業目的은 UNIX의 基本으로 하는 것은 POSIX, 國際標準에의 對應으로서는 X/OPEN, 美國標準局의 Application Portabililty Profile 등의 標準이다.

또한 UNIX의 System과 버클레이판 兩者의 Application Support에 대해서도 IBM의 AIX를 開發의 Base로 하고 있다. X/Open이 國際標準의 設定을 정하는 機關인데 반해 OSF는 새로운 소프트의 開發, 商品化까지 행하는 事業機關이다.

OSF가 追求하는 目標은 ①Multivendor하의 소프트웨어의 共通性 確保 ②異機種컴퓨터 接續 確保 ③Personal Computer 에서 Super Computer까지 接續의 確保이다. 이를 實現하기 위해서는 UNIX의 標準化에 의한 OS統一과 Appliation interface 統一을 기하는 것이다.

(3) 應用 소프트웨어 技術

應用 소프트웨어는 컴퓨터 하드웨어와 시스템 소프트웨어를 利用하여 資料를 處理하는 데 利用되는 業務用 소프트웨어를 말한다. 近來에 들어 人間生活의 어느 한 部分도 컴퓨터와 關聯되지 않은 分野가 없고, 따라서 應用 소프트웨어의 種類는 일일이 列舉할 수 없을 程度로 많아졌다. 企業體의 人事管理, 在庫管理, 經營管理등을 위한 經營情報處理用 소프트웨어, 워드프로세서와 近距離 네트워크를 基盤으로 한 事務自動化(Office Automation) 소프트웨어나 住居環境의 安定性, 便利性, 經濟性을 위한 家庭自動化(Home Automation) 소프트웨어는 물론, 실시간處理 技術의 發展에 따른 工場自動化(Factory Automation) 소프트웨어와 같은 産業用 소프트웨어, 宇宙船의 軌道計算, 航空力學, 地震觀測 등의 科學分野를 위한 소프트웨어 등이 모두 應用 소프트웨어에 屬한다.

應用 소프트웨어의 代表的인 例로 資料處理(Data Processing)를 들 수 있는 데, 이는 컴퓨터 歷史와 함께 始作된 分野로서 現在 流通되고 있는 소프트웨어의 大部分이 이에 屬한다고 볼 수 있다. 從來에는 주로 숫자나 文字 데이터의 處理에 置重하였고, 그나마 모두가 統計處理 등과 같은 算術的 演算을 爲主로 하였다. 近來에는 그와 같은 單純한 算術處理 領域을 벗어나 意思決定을 支援하고 나아가서 組織體를 總括적으로 管理하는 經營情報 시스템으로 發展하고 있다.

CAD/CAM 分野는 最近의 그래픽 技術과 PC CAD의 發達로 인하여 單純한 圖面作成 作業은 價格이 저렴한 PC에서 주로 하며, 概念設計를 위한 解釋 및 試驗과 모델링 機能이 重視되는 CAE(Computer Aided Engineering)는 工學用 워크스테이션을 주로 利用하는 趨勢이다.

從來에는 하드웨어와 소프트웨어를 묶어 툰키베이스로 開發되었으나 最近에는 여러 하드웨어에 移植하여 遂行될 수 있도록 專門開發業體에서 패키지화하여 開發하는 傾向이 있다.

(4) 人工智能 技術

初期의 人工智能 소프트웨어 分野에서는 주로 明白한 推論方法의 構成에 초점을 두고 있었으나 1960年代 以後로는 知識工學의 研究에 重點을 두고 있다. 知識을 表現하고, 그를 效果的으로 使用하며, 知識을 習得하는 方法에 대한 研究가 活潑히 進行되고 있다. 이제까지 成功의 인 人工智能 소프트웨어는 大部分이 知識基盤 專門家 시스템이었으며, 앞으로 實用度가 매우 높아질 것으로 豫想되는 分野는 人工智能 言語와 專門家 시스템, 自然語 處理, 人工視覺, 로보트 등을 들 수 있다. 人工智能 소프트웨어의 基本 構成은 해당 分野의 知識을 인코딩하여 表現하고, 問題에 聯關된 知識을 檢索하는 패턴매칭, 수많은 可能한 解答들 중에서 만족할 만한 解答을 찾아내는 探索過程으로 이루어진다.

人工智能 소프트웨어의 開發을 效果的으로 支援하기 위하여 人工智能 言語와 人工智能 소프트웨어 開發道具에 대한 研究도 活潑해지고 있다. 人工智能 프로그래밍 言語로는 問題志向의 프로그램을 할 수 있고, 問題를 解決하는데 節次的이기 보다는 解決策의 種類를 明示할 수 있어야 하며, 複雜한 問題處理를 自動的으로 同時에 遂行할 수 있는 言語가 必要하다. 이러한 人工智能 言語로는 函數的 言語와 論理言語가 있다. 函數的 言語는 람다(lambda) 計算에 基盤을 두고 函數들을 定義하며, 例로써 LISP, FP, HOPE 등을 들 수 있다. 論理言語는 一次論理에 基盤을 두고 關係를 正義하는 節들의 集合으로 構成되며 Prolog를 例로 들 수 있다.

한편 人工智能 소프트웨어 開發道具로는 주로 專門家 시스템 開發用 패키지를 들 수 있는데 例로써 KEE(Knowledge Engineering Environment)와 ART(Automated Reasoning Tool), OPS5 등이 있다. 앞으로 專門家 시스템이 活性化되려면 보다 넓은 領域의 推論能力을 確保해야 한다. 現在로서는 제일 複雜한 專門家 시스템이라도 대개 3段階 程度의 推論能力 밖에 갖고 있지 못하지만 앞으로는 9~10段階의 推論能力을 가져야만이 高度의 일을 할 수 있을 것이다.

自然語 處理는 사람이 使用하는 言語를 거의 그대로 컴퓨터가 處理하는 技術이다. 機械翻譯이 自然語 處理의 가장 重要한 應用分野로 되고 있으며, 自然語에 의한 質疑應答 시스템, 音聲 認識 시스템, 自然語 인터페이스 등에 대한 研究가 進行중에 있다. 機械翻譯에는 pivot 방식과 transfer 방식이 있다. 前者는 入力言語가 表現하는 意味를 個別言語에 依存하지 않는 中間言語로 記述하여 그 中間言語를 바탕으로 意味가 같은 다른 言語로 出力하는 것이다. 後者는 入力言語의 文法的 構造를 갖는 中間 表現을 만들어 이와 對應하는 다른 言語의 文法的 構造로 變換하고 이어서 그言語를 만들어 내는 방식이다. 機械翻譯 技術 水準은 아직 對象 文法과 分野를 制限하고 있는 程度이며 翻譯 速度와 質의 向上을 위해서는 高速處理, 並列處理와 같은

새로운 處理技術과 翻譯을 위한 辭典, 言語를 나타내는 概念의 體系와 概念을 說明하는 知識베이스 및 意味 네트워크 構築 등의 人工智能 技術이 必要하게 된다.

(5) 開發技術

소프트웨어 費用이 急上昇함에 따라 開發技術의 重要性이 크게 부각되고 있다. 現在의 開發技術 趨勢로는 먼저 既存의 開發方法과 道具들을 基盤으로 有機的인 開發環境을 構築하려는 試圖이 있고, 두번째로는 프로토타이핑, 再使用, 自動化 프로그래밍과 같은 새로운 paradigms에 관한 研究의 두가지를 들 수 있다.

이제까지는 소프트웨어 開發過程의 각 段階別로 特定한 開發方式이나 道具들을 使用해 왔다. 近來에 들어서는 特定 方法이나 道具의 支援만으로는 效果的인 소프트웨어開發이 어렵다는 것을 認識하게 됨에 따라 開發方法, 開發道具 및 開發管理 節次 등을 有機的으로 統合한 소프트웨어 開發環境을 構築함으로써 開發過程의 生産성과 品質의 向上을 꾀하고 있다. 이러한 開發環境으로는 既存의 道具들을 結合해 使用하면서 開發管理에 重点을 두고 있는 SDEM/SDSS, 特定 프로그래밍 言語를 基盤으로 한 APSE(Ada Programming Support Environment) 및 INTERLISP, SMALLTALK 등이 있으며, 開發方法을 基盤으로 한 環境으로는 AIDES, IDE 등이 있다. 특히 APSE는 Ada 言語를 基盤으로 하고 있으며, 프로젝트 라이프사이클을 통해 각 프로젝트와 關聯된 情報을 保管하는 데이터베이스와 인터페이스, 開發支援道具 세트로 構成되어 있다. 또한 소프트웨어 팩토리(Software Factory)에서는 소프트웨어 開發 全 過程을 網羅하는 包括的인 環境을 提供한다. 여기서는 開發方法, 開發道具들을 모아 共通的인 開發環境을 構成함으로써 一貫된 開發 工程을 提供하여 生産성을 높이고, 소프트웨어 開發에 關聯된 情報을 共有하여 重複開發에 따른 非能率을 防止하기도 한다.

이러한 開發環境 構築외에도 새로운 開發 paradigms을 導入함으로써 소프트웨어의 品質과 生産성을 向上시키려는 努力이 꾸준히 展開되고 있다. 새로운 paradigms으로는 再使用 技法, 프로토타이핑, 自動化 프로그래밍을 들 수 있다.

소프트웨어 再使用이란 소프트웨어 開發에 關聯된 過去의 經驗이나 資料를 새로운 開發 狀況에 알맞게 適用시키는 것을 말한다. 過去의 비슷한 開發經驗이나 그에 關聯된 프로그램 및 資料를 再使用함으로써 開發期間을 短縮하고 所要人力을 줄일 수 있을 뿐만 아니라, 이미 充分한 테스트를 거친 소프트웨어를 다시 使用함으로써 새로운 소프트웨어의 信賴性이나 維持補修성을 크게 向上시킬 수 있다. 소프트웨어의 再使用은 그 대상에 따라 빌딩블록 再使用과 패턴 再使用으로 나눌 수 있다. 빌딩블록 再使用은 소프트웨어 빌딩블록들을 라이브러리에 모아놓고 새로운 소프트웨어의 開發에 필요한 빌딩블록을 찾아내 結合시켜 나가는 方法이다. 例로서 RSL

(Reusable Software Library)에서는 데이터베이스 質疑를 통해 라이브러리로 부터 必要한 컴포넌트나 혹은 그와 類似한 컴포넌트에 대한 여러가지 有用한 情報를 提供받을 수 있으며, 再使用 라이브러리를 소프트웨어 設計段階와 連

結하여 使用함으로써 再使用 成果를 더욱 높이고 있다. 이에 비해 패턴 再使用은 目標하는 소프트웨어에 대한 一般의인 模型을 만들어 놓고, 거기에 適切한 媒介變數를 適用함으로써 必要한 소프트웨어를 生産해내는 積極的인 方法이다. 패턴 再私用의 例로는 Application Generator 등을 들 수 있는데 아직은 그 應用分野가 限定되어 있다. 이외에도 日本의 시그마 프로젝트에서는 汎國家的인 共同 開發環境을 助成하여 소프트웨어의 再使用을 活性化시키고 있으며, 소프트웨어 팩토리 등과 같은 開發環境에서도 소프트웨어 情報의 共有와 再使用을 支援하고 있다. 또한 近來에는 소프트웨어 開發에 抽象化(abstraction) 技法과 個體指向 프로그래밍(object oriented programming) 技法의 使用이 勸獎됨에 따라 소프트웨어 再使用의 有用성과 實用化 可能性이 더욱 커지고 있다.

한편, 컴퓨터나 關聯 資源의 費用이 매우 비싸던 時期에는 소프트웨어 開發過程에서 컴퓨터 資源을 效果的으로 活用키 위해 具現을 可能한 뒤로 미루고 要求分析이나 設計 등의 計劃段階를 매우 慎重하게 遂行하는 傳統의인 循期開發 方法을 사용하였다. 그러나, 근래에 들어 소프트웨어 開發에 使用되는 컴퓨터 資源의 價格 下落과 性能 向上으로 經濟的인 妥當性보다는 最終 開發製品의 品質이 보다 더 重要해졌고, 開發에 長期間이 所要되는 循期開發 方法보다는 可能한 빨리 使用者의 要求를 滿足하는지를 確認할 수 있는 方法이 必要해짐에 따라 프로토타이핑 技法에 대한 關心이 고조되고 있다. 프로토타이핑은 迅速한 開發道具를 利用할 수 있는 環境에서 小規模 人力과 短期間에 걸쳐 試製品을 만들어 使用者에게 開發하려는 소프트웨어의 動作狀態를 보여줌으로써 使用者의 要求를 正確히 把握하여 本 開發로 들어가는 것이다. 프로토타이핑에서는 試製品을 最終製品으로 成長시킬 수도 있고, 試製品을 要求分析明細나 設計明細로 採擇한 뒤 傳統 循期開發 方法에 의해 開發을 進行할 수도 있다.

소프트웨어 開發 自動化란 소프트웨어 開發의 一部 내지 全 工程을 종래의 手作業 대신에 컴퓨터를 利用하는 것을 말하며, 이를 CASE(Computer Aided Software Engineering)라고 부르기도 한다. 시스템 分析과 設計에 必要한 各種 設計圖面의 作成을 CAD의 原理로 自動化하고, 이를 토대로 CAM의 原理에 입각, 原始코드를 自動生産하는 것이다. 소프트웨어 自動化的인 窮極的인 目標은 소프트웨어 壽命週期 全般에 걸쳐 一貫된 自動化 環境을 提供하는 것으로 이를 위해서는 效果的인 開發支援道具의 開發과 充分한 技術蓄積이 必要하다. 自動化에 必要한 道具로는 設計 支援道具, 프로그래밍 支援道具, 테스트 支援道具, 開發管理 支援道具등을 들 수 있으며, 소프트웨어 開發者는 이러한 道具를 利用하여 컴퓨터 단말기에서 對話 形式으로 作業을 하게 된다. 現在의 技術水準은 소프트웨어 開發工程 單位로 自動化가 이루어진 程度이며,

大部分 프로그래밍과 테스트 段階에 制限的으로 適用되고 있다. 따라서 시스템 分析에서 부터 原始 프로그램의 自動 生産까지 取扱하는 一貫된 道具가 없어 使用者는 多様な 供給先에 依存 하게 되어 각종 異機種 道具들의 統合과 互換性이 問題가 되고 있다. 소프트웨어 開發의 自動 化는 크게 두가지로 나누어지는 데 하나는 VHLL(Very High Level Language)의 범주에 속 하는 것이고, 다른 하나는 知識베이스를 使用하는 것이다. VHLL方式에서는 소프트웨어의 節次 的인 設計나 프로그래밍의 상세한 事項에 구애됨이 없이 問題指向的 構造를 갖는 言語인 VHLL 을 利用하여 소프트웨어 모델을 表現한 후 言語處理機를 통하여 目的하는 소프트웨어 시스템 을 얻게 되며, 知識베이스를 利用하는 方式에서는 시스템내의 知識을 사용하여 使用者 要求의 서는 시스템내의 知識을 利用하여 使用者 要求의 不安定性, 矛盾性, 不明確性 등을 保全하면서 要求의 意味를 解釋하여 내부에서 일단 技能 사양을 變換한 후, 다시 應用分野의 知識과 프로그래밍 知識을 利用하여 目標한 프로그램과 데이터群으로 最終 變換하는 接近方式이다.

(6) 品質保證 技術

소프트웨어의 使用領域이 擴散됨에 따라 그 品質이 産業社會 全體의 質을 左右하게 되었다. 이에 따라 소프트웨어가 規格화된 技術要求 條件을 滿足하면서 充分한 信賴性을 提供할 수 있도록 하는 데 必要한 事項들을 事前에 計劃하고 組織化시켜 소프트웨어 開發過程을 支援하는 品質保證 技術의 重要性이 날로 커지고 있다. 소프트웨어 品質保證 活動은 프로그램의 적정성 與否를 客觀的으로 判斷할 수 있도록 하며, 프로그램이 갖추어야 할 最小限의 基準을 提示하여 소프트웨어의 委託開發과 流通促進을 誘導하는 한편, 소프트웨어의 質的인 向上을 促進하여 使用者를 保護하려는 方向으로 展開되고 있다. 특히 소프트웨어 製品 자체에 대한 品質評價 뿐만 아니라 開發過程에서 使用되는 方法論, 支援道具, 文書化 作業은 勿論 標準化에 이르기까지 開發에 關聯된 모든 事項에 대한 品質을 改善하려는 努力을 展開하고 있다. 大規模 소프트웨어에 는 수 많은 開發요원들이 投入되는 데 開發요원들 간의 意思疏通이 제대로 이루어져야 만이 適期에 滿足할 만한 品質의 소프트웨어를 開發할 수 있으며, 이를 위해서는 標準化가 필수적이다. 소프트웨어에 대한 標準化의 內容으로는 開發過程의 각 工程別 作業 內容, 作業 順序, 工程 간의 關係, 開發文書의 樣式과 內容, 用語의 選擇과 使用, 表記法 등이 包含되고 있다. 現在는 각 業體別로 나름대로의 開發環境에 알맞는 標準化 技法을 保有하고 있는 데, 예를 들면 日本 후지쓰의 SDEM이나 히다찌의 SPDS가 있으며, 이로 인하여 각 業體는 20~50%까지, 또는 工程에 따라서는 數倍의 生産性 向上을 達成하고 있다.

마지막으로 最近에 들어 컴퓨터 네트워크의 使用이 活潑해짐에 따라 데이터의 流過程에서 個人 및 社會 각 分野에 커다란 影響을 미칠 수 있는 情報의 不法的인 盜難이나 流出을 막기 위한 保護 對策이 重要시 되고 있다. 保護對策으로는 패스워드등을 使用하여 情報에 대한 物理

的 接近을 制限하는 一차적인 方法과 一차적인 保護措置가 破壞되었을 때에도 情報를 安全하게 保護할 수 있도록 技術的으로 情報를 暗號化하는 一차적인 方法이 있다. OSI에서는 情報를 暗號化하고, 祕密 키(private key)를 통해 상대방이 正確히 認許된 뒤에야 情報를 解讀할 수 있도록 하며, 특정 資源에 대한 액세스를 統制하고, 安全性 達成을 위해 有效한 轉送經路를 選擇하도록 하는 것등을 내용으로 하는 保安構造를 提案하고 있다.

나 各國의 技術開發 動向

世界 各國은 情報化 司會의 先頭 走者로 나서기 위하여 소프트웨어 技術開發 事業에 注力하고 있다. 近來에는 注文에 의한 프로그램 開發이 줄어들고 패키지 形態의 汎用 프로그램이 점차로 많이 普及되고 있으며, 技術的인 면에서는 生産技術의 提高를 위한 標準化 技術, 再使用 技術, 自動化 技術등이 重點的으로 開發되고 있으며, 人工智能 技術의 適用化 趨勢에 따라 第 5世代 컴퓨터의 基盤技術 開發에도 많은 努力을 기울이고 있다. 主要 國家別 技術開發 프로젝트 現況을 整理하면 <圖表 II -2-302>와 같다.

<圖表 II -2-302> 國家別 소프트웨어 技術開發 計劃

國 家	프로젝트	技 術 開 發 內 譯
美 國	Ada 開發 SCI SDI MCC SEI STARS	Ada 實用化 및 프로그램 開發環境 助成 專門家 시스템, 知識基盤시스템 등 人工智能 시스템 開發 知識情報處理시스템 開發 半導體 및 소프트웨어 技術 開發 再使用, 自動化 등 소프트웨어 工學 技術 開發 軍사용 소프트웨어 開發
日 本	FGCS SIGMA INS	第 5世代 컴퓨터 및 소프트웨어 技術 開發 소프트웨어 生産工業化시스템 構築 綜合情報通信網 構築
유 럽	ESPRIT EUREKA RACE EJOB	소프트웨어, 高度情報處理, OA, CIM 등 電子 및 情報 關聯 尖端技術 共同開發 및 産業化 유럽 廣帶域通信網 普及 光 論理소자 및 컴퓨터 開發
英 國	ALVEY PAIT IDA	政府, 産, 學, 研 共同으로 情報技術 開發 情報産業 全般에 대한 育成計劃 綜合情報通信網 構築
프 랑 스	CONCERTO	소프트웨어 開發環境 構築

(1) 美 國

美國의 소프트웨어 技術 開發은 Ada 開發計劃과 戰略情報處理計劃(SCI : Strategic Computing Initiative), 戰略防衛構想計劃(SDI : Strategic Defence Initiative), MCC(Microelectronics & Computer Technology Corp.)의 開發計劃, SEI(Software Engineering Institute)의 開發計劃 등을 中心으로 遂行되고 있다.

Ada 開發計劃은 美國防省에서 소프트웨어 開發의 標準化와 效率化를 위하여 1975년부터 高級 프로그래밍 言語인 Ada의 開發을 目標로 推進되었으며, 1983년 以後부터는 Ada의 充實化와 利用促進을 위해 1990년 까지 第 2段階 計劃이 推進되고 있는데 주요 內容으로는 國防省 委託 프로젝트의 Ada使用 義務化, 소프트웨어 管理指針 設定, 각종 Ada 프로그램의 目錄化, 라이브러리 프로그램, 어플리케이션 제너레이터, 知識基盤 시스템 등 Ada 소프트웨어의 委託 開發 등이 包含되어 있다.

SCI는 美國防省을 中心으로 한 大規模의 컴퓨터 開發 프로젝트로서 專門家 시스템, 人工智能을 利用한 知識 시스템의 開發을 目標로 하며, 初期段階에서는 開發支援道具와 試製品 開發 등을 통해 基盤을 整備하고 後半에는 人工智能 등의 技術基盤을 構築하여 軍事用 어플리케이션을 開發할 豫定이다.

SDI는 1985년부터 國防省이 戰略彈道미사일 對應策으로 推進하고 있는 計劃으로 超高速 컴퓨터, 知識情報處理 시스템 등의 開發計劃을 包含하고 있다.

MCC는 獨占規制法에 의해 1983년 컴퓨터 製造業體 등이 合作하여 組織한 尖端技術 研究機關으로 주로 半導體의 패키징 및 接續技術, 並列處理 컴퓨터 構造, 데이터베이스 시스템, 人工智能/知識基盤 시스템, 그리고 大規模 소프트웨어 開發의 生産性 向上 및 設計技術의 開發 등을 目標로 하고 있다.

SEI는 카네기멜론大學의 소프트웨어 工學 研究所로 向後 國防省에서 使用하는 컴퓨터시스템의 信賴性 向上 및 시스템 應用技術 등의 開發을 推進하게 될 것이다.

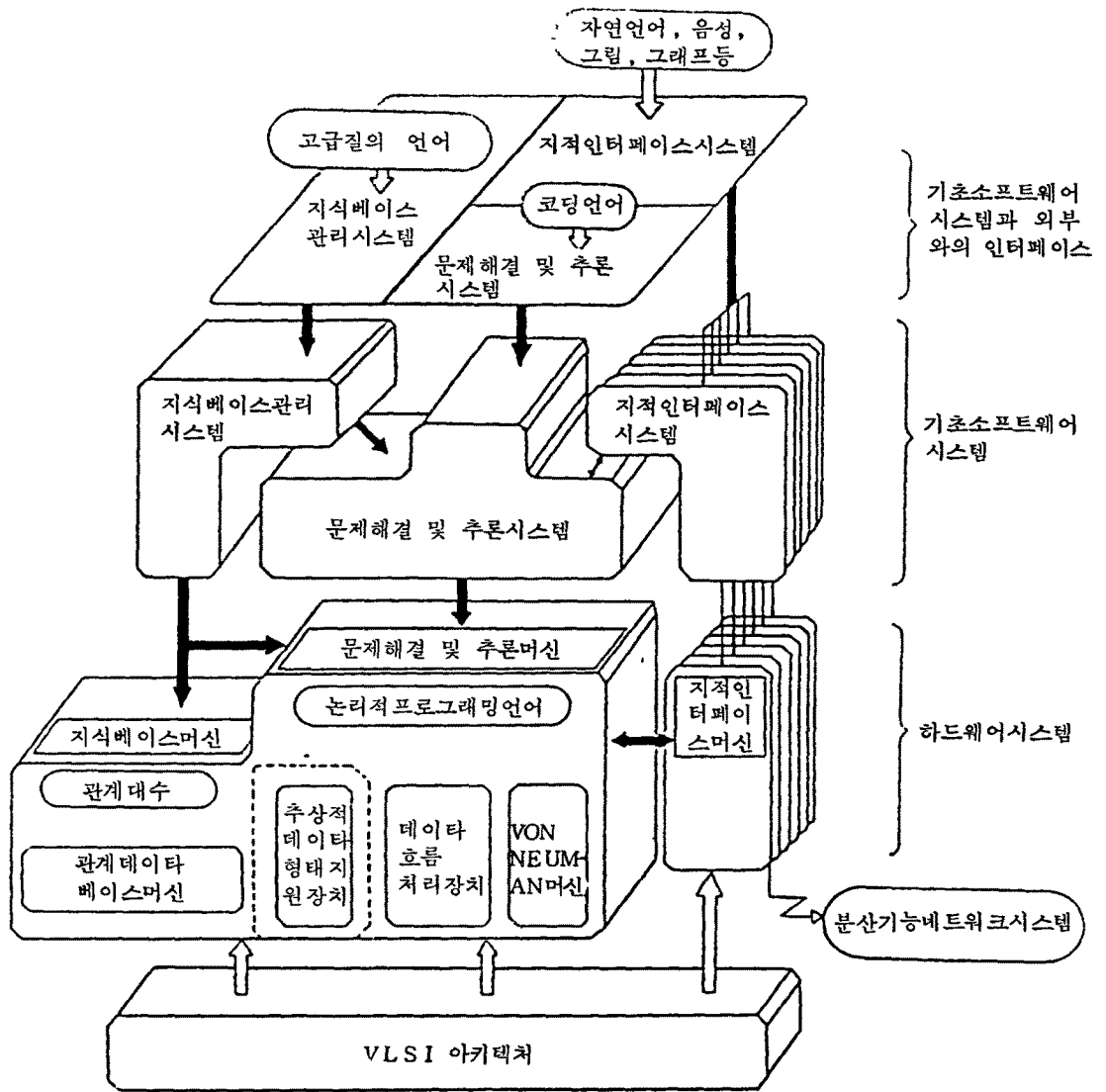
(2) 日 本

日本은 開發費가 크고 開發에 많은 時間을 所要하는 高度의 汎用 프로그램과 中小企業型 汎用 프로그램, CAI를 中心으로 하는 教育, 學習用 프로그램의 開發 및 流通을 推進하고 있으며, 企業의 技術開發 能力 向上을 目的으로 하는 特定 프로그램 委託開發制度를 運營하고 있다. 소프트웨어 開發을 위한 自動化支援道具의 開發 및 普及과 소프트웨어 開發環境의 統合技術을 推進함으로써 生産性 및 信賴性이 크게 向上될 것으로 期待된다.

第 5세대 컴퓨터(FGCS : Fifth Generation Computer System) 開發計劃에서는 컴퓨터 自身이 문제 解決方法을 追跡하고 決論을 내릴 수 있는 推論技能, 知識基盤管理 技能, 知的 인터페

<圖表 II -2-303>

5세대 컴퓨터 시스템에 대한 基礎構成圖



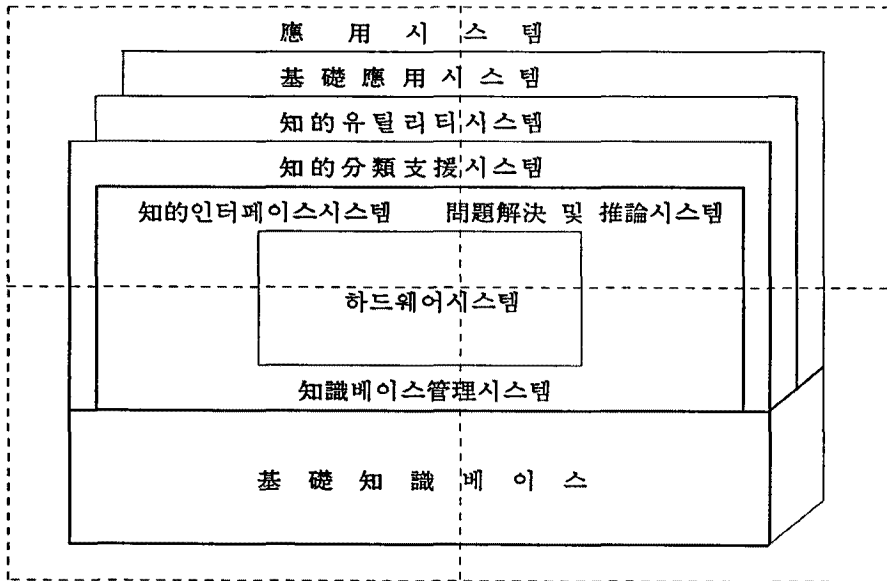
자료 : KeyNote speech : challenge for knowledge Information Processing System

이스 技能, 自動 프로그래밍 등의 소프트웨어 技術에 대한 研究開發을 進行하고 있다.

소프트웨어 生産工業化시스템(SIGMA : Software Industrialized Generator & Maintenance Aids) 開發計劃에서는 소프트웨어 開發의 生産性 및 信賴性을 向上시킨다는 目標하에 開發의 全 工程에 컴퓨터를 導入하여 自動化하기 위한 것으로 하드웨어에 獨立的인 標準 소프트웨어 開發環境 構築, 프로그램이나 技術情報 등의 檢索 및 傳送을 위한 네트워크 시스템의 確立 등을 追求하고 있다. 시그마시스템이 提供하는 서비스는 소프트웨어 開發支援, 데이터베이스, 通信서비스로 나누어지며 소프트웨어 産業의 地域隔差를 解消하는 데도 도움이 될 것으로 豫想 된다.

이외에도 電子計算機 相互運用 데이터베이스 시스템은 文字는 勿論 圖形, 畫像, 音聲 등의 멀티미디어 情報를 分散데이터베이스와 네트워크를 통해 自動 檢索할 수 있도록 할 豫定이며, NTT에서 推進하고 있는 INS(Information Network System)는 多樣한 情報에 대한 通信 서비스를 하나의 通信網을 통해 저렴한 料金으로 提供함으로써 情報社會의 通信基盤構造를 마련 하려 하고 있다.

<圖表 II-2-304> 5세대 컴퓨터의 소프트웨어시스템의 構成



資料 : KEYNOTE SPEECH

(3) 유 럽

統一된 유럽 建設이라는 最終目標을 達成키 爲해 發足한 歐洲共同體(EC)는 1980년대에 들어면서 技術向上과 國際競爭力의 強化를 爲해 컴퓨터關聯 情報産業育成政策을 多角度도 推進하고 있다.

유럽 情報技術研究開發 戰略計劃(ESPRIT : The European Strategic Programme for R&D in Information Technology)을 통해 소프트웨어 生産自動化 技術을 開發하고, 專門家 시스템, 事務自動化, CIM(Computer Integrated Manufacturing) 등에 대한 研究開發을 推進하고 있다.

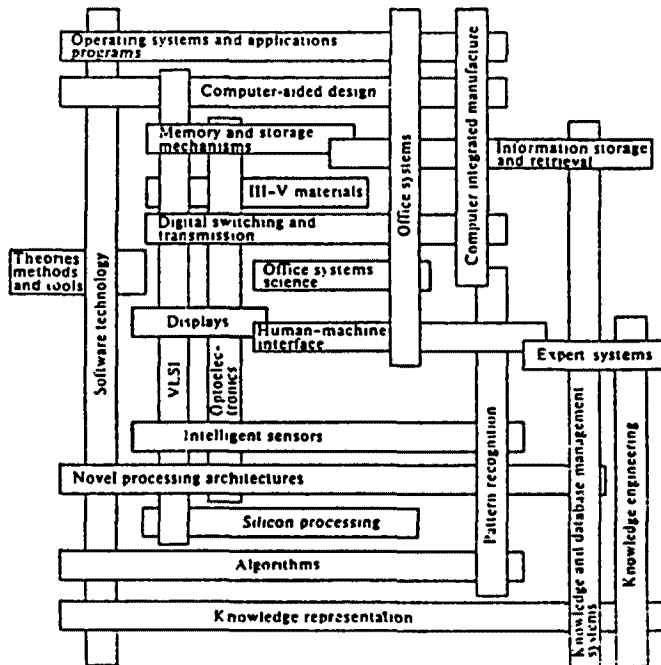
또한 ESPRIT計劃이 基礎研究인데 비하여 EURECA(European Research Coordination Action)計劃에서는 情報産業 技術의 商品化를 目標로 하여 企業體와 研究機關의 協力體制를 強化함으로써 高度 科學技術 分野에 대한 産業 生産性 및 國際競爭力을 높이기 爲해 努力하고 있다.

이외에도 RACE(R&D in Advanced Communications-Technologies in Europe)計劃에서는 유럽 共通의 디지털 通信網을 構築하고, EJOB 計劃(European Joint Optical Bistability Project)에서는 光컴퓨터시스템의 開發을 推進중이다. 이와 아울러 유럽에서는 機械翻譯 시스템의 研究 開發도 活潑히 進行되고 있다.

특히 英國은 SPS(Software Product Scheme) 등을 통해 소프트웨어 패키지의 開發 및 普及

<圖表 II -2-305>

ESPRIT計劃의 概要와 相互關係



資料 : Oxford Surveys in Information Technology

을 促進시키기 위한 政策的인 支援을 하고 있으며, ALVEY 프로그램을 통해 소프트웨어 工學의 여러 분야와 맨-머신 인터페이스, 知識推論 시스템을 研究開發하고 있다. 또한 企業體에서는 CAD등과 같은 專門分野에 主眼點을 두고 있다.

프랑스는 일렉트로닉스 5개년 計劃에서 컴퓨터에 의한 VLSI 設計, CAD/CAM, 소프트웨어 工學, 機械翻譯, 컴퓨터에 의한 敎育 등에 대한 技術을 開發하고 있으며, 특히 마이크로프로세서를 利用한 自動化 技術에 큰 成果를 보이고 있다.

<圖表 II-2-306> 主要 先進國들이 支援하는 소프트웨어 關聯 R&D 프로젝트

國名	프로젝트	期間	R&D의 重點內容	資 金
美 國	STARS(Software Technology for Adaptable and Reliable Systems) ...國防省	FY1984-1990	統合되고 自動化된 소프트웨어디자인 Tool. Software engineering institutes.	FY 1984년에 500萬달러
	Strategic Computing 國防省	FY1984-1994	Multi-parallel processing. 人工知能用 S/W(speech와 自然言語理解: vision and expert system)	H/W와 S/W R&D를 하는 프로젝트 全期間에 걸쳐 7億5,000萬달러에서 10億달러
	Space Station NASA	FY1984-1994	Network operating systems와 S/W tools	Space station H/W와 S/W의 開發을 위한 프로젝트 全期間에 걸쳐서 80億달러
日 本	超高速컴퓨터시스템	1981-1988	超大型컴퓨터 O/S와 프로그래밍 言語	H/W, S/W開發을 위해 全期間에 걸쳐 2億달러
	5세대컴퓨터시스템	1982-1991	S/W engineering과 人工知能이 프로젝트의 重要部分: 目標은 K I P (Knowledge Information Processing) machine	H/W, S/W開發을 위해 全期間에 걸쳐 4億 5,000萬달러
프 랑 스	Electronics Plan/Software 5個年計劃	1983-1987	Software Engineering	컴퓨터서비스와 S/W 開發을 위해 1987년에 프랑스政府로부터 6億8,000萬달러
	Marisis	1983-1986	超大型컴퓨터用 S/W	H/W와 S/W開發을 위해 1986년에 프랑스政府로부터 1,500萬달러
英 國	Alvey 소프트웨어 프로그램	1983-1988	Software Engineering	S/W開發을 위해 全期間에 걸쳐 1億1,400萬달러
유 공 동 체	ESPRIT(European Strategic Program for Research and Development in Information Technology)	1984-1988	S/W engineering과 人工知能	全期間에 걸쳐 S/W technology 프로그램을 위해 1億달러, EC가 ESPRIT에 提供하는 總額數는 6億4,500萬달러

資料) A Competitive Assessment of the United States Software Industry
 註) FY는 Fiscal Year(會計年度)로 美國의 會計年度는 6月 30日로 끝남.