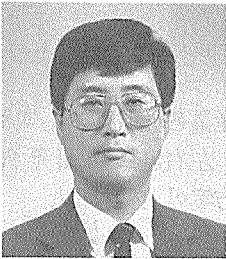


先進國의 소프트웨어 開發動向



崔 相 鉉

金oble 소프트웨어(주) 研究所長/理事

소프트웨어의 표준화 부품화와 함께 또다른 하나의 추세는 소프트웨어의 거대화이나 선진국에서는 그동안 수많은 소프트웨어가 개발되어 운영되어 왔으며 이들의 사용 경험을 바탕으로 사용자들은 더욱 수준이 높은 서비스를 요구하고 있다.

1. 부드러운 소프트웨어

컴퓨터가 처음으로 등장한 1950年代에의 컴퓨터는 별도의 건물에 있는 컴퓨터 센터에서少數의 科學者들이 使用하는 것이었다. '60年代가 되면 일반 건물에 별도의 施設이 갖춰진 電算室에서 選拔된 電算要員이 使用하는 것이었으며, '80年代에서는 일반 事務室에서 職場人이 使用하게 되었고, 이제는 점차 어느곳에서 누구라도 쉽게 使用할 수 있도록 發展되어 가고 있다.

사람의 말이나 목소리를 理解하게 하는 소프트웨어, 종이에 인쇄된 文字 또는 손으로 쓴 글자를 認識하는 소프트웨어 등은 딱딱하고 쓰기 불편하던 컴퓨터를 부드럽고 쓰기 편한 道具로 만들어 專門家가 아닌 누구라도 손쉽게 컴퓨터를 쓸 수 있게 할 것이다. 딱딱한 컴퓨터를 부드럽게 만드는 소프트웨어는 使用者 측에서 보면 Data의 入力 方法, 시스템의 指示 方法 등 주로 User Interface에 集中되어 있는 것처럼 보인다. 그러나 다른 곳에 있는 데이터를 가지고 온다든가, 또는 다른 시스템에서 開發된 프로그램을 使用한다든가 하여 使用 形態가 조금만 複雜하게 되면 시스템간의 統一性의 缺如로 使用 方法이 數十倍 以上으로 複雜해 지는 것이 보통이므로, 부드러운 소프트웨어는 소프트웨어 전반에 걸친 標準化를 要求하고 있으며, 또한 그러한 標準化가 여러 分野에서 進行 되고 있다.

소프트웨어가 부드럽다는 表現에는 使用者가 쓰기 부드러운 機能을 提供한다는 意味外에 또 다른 한가지 뜻이 있다. 하드웨어는 한번 製造 되면 機能을 變更한다든가 修正을 하기가 거의 不可能 한데 비해 소프트웨어는 쉽게 바꿀 수 있다고 하여 소프트웨어라는 名稱이 붙여졌다. 소프트웨어의 初期에서는 확실히 그러했으며,



표준화를 위한 또 하나의 큰 흐름은 산업계 및 국제 표준 Interface를 확립하는 일이다.

지금도 소프트웨어의 규모가 작을 때에는 아직도 그 말은 틀리지 않는다. 그러나 소프트웨어의 규모가 크고 복잡해지고, 다른 소프트웨어들과 연동이 되기 시작하면 아무리 소프트웨어라고 해도 좀처럼 쉽게 고칠 수 없게 된다. 그런 면에서는 하드웨어보다 더 딱딱하고融通性이 없는 것이 既存의 소프트웨어라고 할 수 있으며, 여기에서 派生되는 損失은 어마어마하게 많다. 이러한 缺點을 補完하기 위해 소프트웨어에 融通性을 賦與한 것이 부드러운 소프트웨어이며, 이는 여러分野의 소프트웨어의 標準化, 部品化를 통해 이루어 지고 있다.

이와 같이 先進國의 소프트웨어는 부드러운 소프트웨어로 發展되고 있으며, 다음에서 그 具體的인 例들을 보기로 하겠다.

2. 計算機에서 事務器機로 變身

英國의 Xerox 社가 컴퓨터를 使用하여 文書 作成을 편하게 할 수 있도록 書面上에 鉛筆, 지우개, 計算機 등을 標示한 Star라는 컴퓨터를 發賣한 것이 1981년이었다. 컴퓨터가 計算 爲主의 機械에서 事務用 器機로 轉換하는 始發點이라고 하겠다. 그 후 이러한 움직임은 Apple 社의 Machintosh로 이어졌고 個人用 PC의 補給

과 함께 더욱 擴大되어 Wordprocessor, Spread Sheet와 같은 事務用 소프트웨어들이 多數 登場하게 되었다.

컴퓨터를 性能 좋은 事務用器機로 쓸 수 있게 해주는 소프트웨어는 더욱 發展하고 있어 事務室에서 日常的으로 쓰고 있는 電話라든가 팩시밀리, OHP, 슬라이드 投影機, VTR 등의 機能을 대신해 주는 소프트웨어가 登場하고 있으며, 取扱할 수 있는 Data는 文字 以外에 畫象이나 音樂도 포함하고 있다. 文字 以外에 映像이나 音樂을 다룰 수 있는 Multi Media 시스템의 登場은 컴퓨터를 더욱 쓰기 쉽게 만들었으며, 이를 使用한 새로운 소프트웨어들이 開發되기 시작하고 있다.

예를 들어 Apple 社는 1988年 末부터 自社의 營業社員 및 技術者에 대한 教育에 Multi Media 시스템을 使用하고 있다. 製品의 紹介 Manual과 營業에 必要한 情報 등을 Video Disk (VD) 3장과 CD-ROM 1장에 넣어두고, VD를 Machintosh의 Hyper Card로 조작하면서 製品의 特徵, 設置方法, 保守方法, 營業方法 등을 對話式으로 學習해 나아간다. 市販用으로는 베토벤의 제9번 교향곡, 고희(以上 美國 Voyager 社) 作業 工程管理(Auther Anderson 社) 마야 文明의 崩壞, 트럭의 運轉練習 등이 있다.

Multi-Media 시스템은 教育·訓練, 販賣促進, 電子出版 Game과 같은 分野에서 상당히 功력을 發揮할 수 있다. 이들 分野의 共通된 特徵은 Multi-Media 시스템을 採用하면서 既存의 方法보다 效果가 더 높아지거나 높아질 可能性이 많다는 것이다.

예를 들어 本社에 있는 Data-Base에서 손님이 원하는 物件의 最新 映像을 불러내어 즉시 보여준다든가, 教科書에서 필요한 部分을 拔萃해 읽다가 計算式이 있으면 그 식을 Simulation해 본다든가 하는 식으로 종이나 책으로 不可能한 일을 할 수 있는 分野이다. 다시 말해 計算機의 演算能力, 大容量 記憶裝置, Network 機能 등을 最大로 活用하여 時間과 距離의 制約을 없애므로서 效果를 높일 수 있는 곳이다.

3. 標準化와 部品化

지금까지의 많은 소프트웨어들은 機種이 바뀌면 動作하지 않는 경우가 大部分이었고, 動作한다고 해도 同一會社의 컴퓨터에 限定 되어 있었다. 심한 경우에는 同一 會社 製品에서도 互換性이 없었는데 IBM이 그 代表的인 例로서 IBM PC에서 開發된 소프트웨어는 IBM 大型機에서는 動作하지 않는다. 따라서 使用者는 필요한 機種別로 소프트웨어를 다시 開發해야 하며, 패키지 소프트웨어를 開發해도 市場이 限定되어 優秀한 패키지 소프트웨어가 出現하기 어렵다. 이러한 不便을 없애고 소프트웨어 開發의 效率성을 높이기 위한 具體的인 움직임이 '80年代後半부터 始作된 應用 프로그램과 컴퓨터 사이의 Interface(API: Application Program Interface)의 標準化이다.

標準化의 큰 흐름중 하나는 大型 컴퓨터 製造會社가 自社의 컴퓨터 사이의 API를 統一하는 作業으로 IBM의 SAA(System Application Architecture)와 日本 富士通의 SIA(System Intergration Architecture)가 그 代表的인 例이다. 지금까지 각 컴퓨터 製造 會社들은 大型機, 中型機, PC 등에 각각 다른 Interface를 적용하여 왔으나, 이들 Interface 위에 새로운 共通

의 Interface를 附加하여 各 機種간의 Interface를 統一하고 있다.

標準化를 위한 또 하나의 큰 흐름은 產業界 및 國際 標準 Interface를 確立하는 일로 OSI(Open System Interconnection)나 UNIX 등이 代表的인 例이다. UNIX는 1969년에 AT & T社의 Bell 研究所에서 開發되었으나 그 후 분화를 계속하여 AT & T版, 버클리 大學版, Sun Microsystem 社 版 등이 병존하고 있다. 이것을 다시 하나로 統一하기 위하여 AT & T와 Sun이 Unix의 統合 計劃을 發表했으며 ISO의 POSIX와 같은 標準案이 나오고 있다.

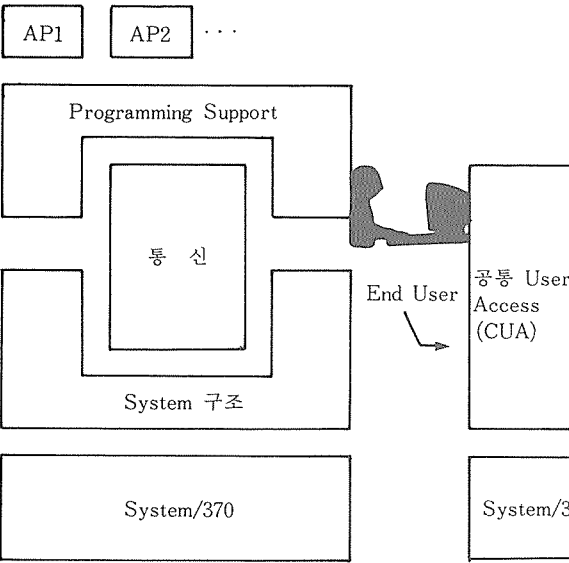
가. IBM의 SAA

API(Application Program Interface)의 標準化에 가장 큰 影響을 미친 것은 '89년에 發表된 IBM의 SAA이다. SAA의 發表에 충격을 받은 大型 컴퓨터 製造會社들은 연이어 SIA(富士通), HIA(日立), DISA(NEC)와 같은 自社의 API를 發表하였다.

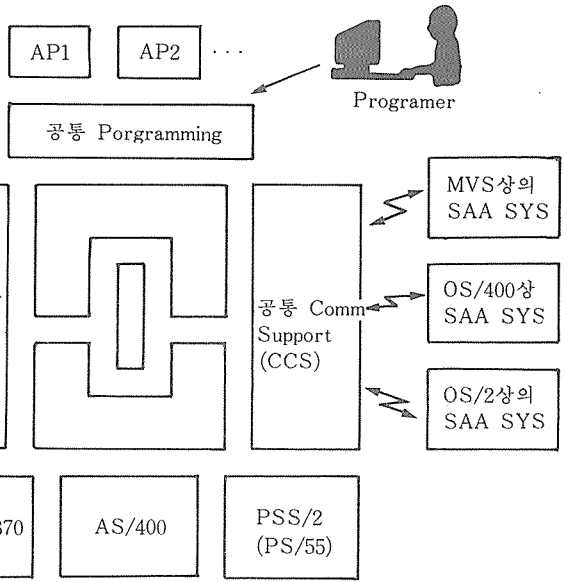
SAA가 지향하는 것은 汎用 大型機(System 370), 中型機(AS/400) 및 워크스테이션(PS/2)에 共通의으로 적용되는 統一된 開發環境이다. 따라서 既存의 상이점을 보완하기 위해서 操作性, Programming Interface, 通信 Interface 등 각 部分에 仕様을 統一시키고 있다. SAA에 대한 概念을 아래의 그림에 表示하였다. SAA에서는 應用프로그램은 共通 Programming Interface(CPI) 위에서 開發한다. 對象 機種은 M-VS 시스템, VM 시스템, OS/2 어느 것이라도 相關없다. 경우에 따라서는 프로그램을 먼저 開發하고 運用에 適當한 시스템을 나중에 決定할 수도 있다. 이렇게 시스템을 눈에 뜨이지 않게 假想化하는 것이 標準 Interface의 重要한 役割이다. SAA의 當面の 目的은 IBM製品간의 接續이나 장차는 다른 會社의 시스템과 相互接續할 可能性도 있다.

나. UNIX

(a) MVS의 구조



(b) System의 구조



〈그림 - 1〉 SAA의 개념

User Interface를 규정한 공통 User Access(CUA), 공통 Programming Interface(CPI), 통신수단을 규정한 공통 Communication Support(CCS)를 기반으로 하여 공통 Application을 만든다.

Unix의 標準化는 주로 IEEE와 같은 學術團體의 主管으로 POSIX와 같은 國際 標準, X/open과 같은 產業界 標準 仕樣이 制定되었었다. 이러한 움직임을 받아 '87年末에 AT & T 는 Sun Microsystem 社와 함께 統合 UNIX 開發 計劃을 發表했다. 여기에는 UNIX 標準化의 主導權을 確保하겠다는 AT & T의 意志가 강하게 나타나 있어, UNIX에 대한 AT & T의 獨占 지배를 우려한 컴퓨터 製造會社들이 '89年 5月에 OSF라는 開發 團體를 만들어 獨自의 開發 作業을 進行하고 있다.

POSIX, X/open과 같은 UNIX 標準化의 目標은 Source Code 水準의 Program 互換性 確保였다. 그 후 AT & T는 Binary Code 互換의 ABI를 제창했으며, OSF는 機種에 獨立的인 中間 言語(ANDF)를 採擇하고 있다. Source Code 互換은 C言語로 作成된 Program이 互換 可能한 水準으로 AT & T가 정의한 SVID, IEEE의 POSIX, X/open과 같은 標準이 여기에 該當한다. Binary 互換은 정해진 Format에 맞는 Binary Code라면 相互 實行이 可能하도록

하는 것으로 Source 互換의 경우 보다 細細한 部分까지 規格을 統一해야 하며, CPU의 Architecture가 같아야 함은 물론 OS의 Library도 共通으로 갖추고 있어야 한다.

4. S.I(System Integration)

소프트웨어의 標準化, 部品化와 함께 또다른 하나의 趨勢는 소프트웨어의 巨大化이다. 先進國에서는 그동안 수많은 소프트웨어가 開發되어 運營되어 왔으며, 이들의 使用 經驗을 바탕으로 使用者들은 더욱 水準이 높은 서비스를 要求하고 있다. 따라서 소프트웨어는 巨大해지고 있으며, 境遇에 따라서는 單獨 시스템만이 아니라 數十 혹은 數百의 컴퓨터가 連結되어 使用되는 소프트웨어, 觀點을 달리하면 컴퓨터를 하나의 部品처럼 쓴 大規模의 소프트웨어가 登場하고 있다. 이러한 大規模의 소프트웨어에 대한 고객의 要求를 分析하여 소프트웨어는 물론 하드웨어 및 커뮤니케이션에 관한 技術을 統合하여 Total Solution을 提示하고 實際 그러한 시

스텝이 運用될 수 있도록 開發 設置까지 하는 서비스를 SI(System Integration) 事業이라고 부른다.

先進國에서는 SI 市場이 急速히 伸張하고 있어 英國에서는 '87년에 34億\$이었던 SI 市場이 '92년에 120億\$이 될 것으로 推定되며, '87년에 2兆 3,000億円, '95년에 6兆 7,000億円이 豫

想되는 日本의 情報서비스 市場에서도 SI의 比率이 急激히 늘고 있다. 특히 日本의 경우에는 '88년에 通産省이 SI 登録 制度를 실시한 것을 契機로 NTT 데이터通信, 新日鐵 情報 通信과 같은 大型 業體들이 SI 事業에 신규 參與함으로써 産業界의 構造 自體가 再編成되고 있다.

