

운영체제의 장래

高 健

서울대학교 計算統計學科 教授

본고에서는 운영체제(Operating System:OS)의 장래에 대해 간략히 살펴보고자 한다. 우선 OS의 정의부터 소개해보자. OS의 정확한 정의에 대해서는 여러 가지 이견이 있을 수 있으나, 대략 아래와 같이 정의할 수가 있다. OS는 소프트웨어의 일종으로서 여러 하드웨어 기기들을 직접 관리하고, 최종 사용자가 컴퓨터를 쓰기 쉽도록 도와주며, 또 시스템 전체의 효율 및 신뢰도를 향상 시켜주는 역할을 담당한다. 기능별로 보면 OS는 시스템내의 여러 자원들을(하드웨어 및 소프트웨어) 여러 사용자들이 공동사용 할 수 있도록 여건을 조성하여 주고, 복잡한 입출력과 화일처리를 대신 담당하여주며, 최종 사용자에게 사용하기 편리한 여러 기능들을 제공해주며, 오류들을 자동적으로 처리하여 준다.

OS의 과거 발달 역사를 살펴보는 것은 OS의 미래를 예측하는데 있어서 많은 도움이 되리라고 생각한다.

초기 OS의 특징으로서는 첫째 당시엔 CPU, 기억장치 등이 매우 비쌌으므로 주로 이 장치들의 이용효율을 극대화 시키는 것을 OS의 주요한 목표로 삼았었다. 당시에는 비단 OS뿐만이 아니라 계산기 구조 등 여러 분야도 CPU 및 기억장치의 이용효율을 극대화 시키는 쪽으로 노력이 모아졌다.

초기 OS에서 이러한 목적으로 개발된 기술들이 일괄처리(batch processing), 스푼링(spooling), 멀티프로그래밍, 스왑핑(swapping) 등이 그 예들이다. 이러한 기술들은 현재에도 매우 광범위하게 사용되고 있다.

둘째의 특징으로서는, 70년대까지의 OS는 매우 대형이었고, centralized되었고, multimode이었다는 점이다. 대형이라 함은 수백명 이상의 사용자를 위한 OS이었다는 뜻이고, centralized 되었다는 것은 모든 하드웨어와 소프트웨어를 공간적으로 한 장소에 집결시켰다는 뜻이다. multimode라 함은 한 OS가 일괄처리(batch), 실시간 처리(realtime processing) 및 대화

식 처리(interactive processing)를 모두 담당했다는 뜻이다. 과거 OS의 이와같은 특징은 하드웨어 가격이 엄청나게 비쌌다는데에 기인한다. 하드웨어 가격이 매우 비쌌으므로 특정 소수 집단만을 위해서 전용 컴퓨터 및 OS를 만들수는 도저히 없었으므로 당연한 귀결이라 하겠다.

셋째로 과거 OS의 특징을 들어보면 하드웨어와 최종 사용자 사이에 존재하는 OS라는 완충지대가 매우 두터웠다는 점이다. 이 역시 그당시의 OS가 수백명이 공유하며 매우 다양한 기능을 모두 함께 제공하는 multimode란 점을 생각하면 당연한 귀결이었다. 결과적으로 과거의 OS는 그것을 만들고 유지하기에 막대한 노력과 경비가 필요하였다. 70년대까지의 대표적인 OS들은 그것을 만드는데 수천 men-year가 들어가는 것이 보통이었으며, OS가 이렇게 비대해지는 나머지 20~30명의 고급인력이 함께 있어도 OS의 전체 구성을 모두 파악하기가 불가능해지게 되었다. 또 시스템이 커지면 커질수록 그것을 유지하고 보수하는데 드는 시간과 노력이 기하급수적으로 증가하게 되었다.

뿐만아니라 OS가 비대해지자 문제는 최종사용자에게도 생기게 되었다. OS가 너무 비대하고 복잡해지자 그것을 사용하는 언어인 JCL(job control language)도 매우 번거로워졌다. 예를 들자면 한 화일로부터 레코드를 꺼내보는 단순한 동작을 해보기 위해서도 여러 가지 복잡한 파라미터들을 이해해야 했고(예를들면 화일의 구조, 블럭 및 레코드의 구조, 파라미터 등) 그때마다 사용자들은 산같이 쌓인 매뉴얼들을 뒤져야 했다. 따라서 OS를 사용하는 방법을 배우기가 무척 어려웠고, 시간이 오래 걸렸으며, 소프트웨어에 대해 웬만한 기초지식이 없으면 엄두도 낼수 없게된 것이다. (물론 과거의 OS가 모두 부정적인 측면만을 지닌 것은 결코 아니다. OS의 미래를 이야기 하려니까 자연히 과거 OS에서 개선 되어져야 할 측면들만이 부각이 되는 것

이다).

70년대 중반 이후로 집적회로 기술이 주목할만한 속도로 발전함에 따라서 하드웨어의 가격은 급격히 싸지게 되었고 소형화하게 되었다. 그 결과로 컴퓨터는 더욱 널리 사용되었고, 과거에는 과학자, 엔지니어 등에 국한되던 컴퓨터의 이용이 가정주부, 회사 직원등으로 확산되기 시작했고, 이제는 컴퓨터 사용자라는 집단의 대다수가 컴퓨터에 대한 전문적인 지식이 없는 사람들로 바뀌게 되었다. 따라서 미래의 OS는 예전과는 달리 비전문가라도 얼마든지 쉽게 사용할 수 있도록 해야한다는 요구가 증대되게 되었다. 각 명령어를 menu-driven으로(사용자가 각 명령어나 그 파라메타를 이해하고 암기하지 못하더라도 얼마든지 그때그때 화면만을 보고 일을 처리할 수 있도록 하는것) 한다든지, 인공지능의 기법을 이용하여 종래의 까다로운 문법을 가진 명령어 보다 훨씬 편리한 입출력 방식을 개발한든지 하는 것이 미래 OS의 추세가 될 것이다. 또 하드웨어가 점점 싸지고 소형화 함에 따라 예전처럼 centralized된 시스템보다는 데이터가 발생하는 그 장소에 여러 컴퓨터들을 분산운용시키고 그것들을 상호 연결시키는 분산처리 또는 네트워크 운영체제가 더욱 보편화될 것이다. 최근 하드웨어 가격의 하락과 함께 통신비의 가격도 많이 낮아지고 있고 이러한 현상은 분산처리 및 네트워크OS의 발전에 더욱 박차를 가할 것이다. 따라서 예전엔 큰 기관의 여러가지 다양한 일들을 처리하기 위하여 중앙에 거대한 centralized 시스템을 쓰던 것이 점차 수십, 수백개의 전문적인 시스템으로 대체가 되고, 이 각각의 전문적인 시스템들은 소형이어서 개발, 유지 및 이해가 비교적 쉬운 것이며, 각기 맡은 일에는 가장 효율이 높은 OS가 될 것이다. 이 소형이고 전문적인 OS들은 네트워크를 통해 상호 연결되어 total시스템을 구축할 것이다.

최근 발전되는 하드웨어 연관기술중에서 OS에 영향을 미치는 또하나의 요소는 CAD/CAM 기술의 발전이다. 컴퓨터를 설계하고 제조하는 과정에 CAD/CAM 기술을 도입하게되자 새로운 컴퓨터가 등장하는데 걸리는 시간은 훨씬 앞당겨지고 있고 마이크로의 경우는 신제품이 나온 이후로 반년이면 더욱 좋은 제품이 미닌 경우엔 일년내지는 삼년내에 더 좋은 신제품이 등장하는 추세이다. 이러한 현상은 최종사용자에게는 좀 더 자주 하드웨어를 새 기종으로 바꾸는 유혹을 받게됨을 뜻하게 된다.

그러면 이렇게 더 빠르고, 더 값이 싼 하드웨어로 자주 바꾸게 되면 어떠한 영향이 최종사용자에게 미치는

가? 또 미래 OS에는 이것이 무엇을 뜻하는가? 일반적으로 과거의 시스템들은 하드웨어가 바뀌면 소프트웨어(OS, 어셈블리, 등)도 함께 새것으로 바뀌어야 했다. 따라서 기종이 바뀔때마다 사용자는 새로운 시스템의 소프트웨어를 배워야 했다. 과거의 OS들이 그 크기가 매우 비대하고 배우기가 매우 어려웠던점을 감안한다면 이것은 꽤 고통스럽고 큰 부담을 사용자에게 안겨주었음이 틀림없다. 또 하드웨어가 바뀌고 시스템 소프트웨어가 바뀌면, 그 기관에서 종래 개발되어 사용중이던 모든 소프트웨어도 새로운 기종에 맞도록 수정하여야 한다, 이제 웬만한 기관이면 그 기관의 업무를 일, 이십년동안 전산화 시키어 왔을터이고 따라서 하드웨어가 바뀔때마다 수정을 가해야 하는 소프트웨어의 양은 엄청나게 늘어났다. 경우에 따라서는 빠르고 값이 싼 새 하드웨어로 대체하는 데서 오는 이익이, 이와같이 소프트웨어를 수정하고, 직원들에게 새로운 소프트웨어를 가르치는데 드는 경비보다 훨씬 못미치게 되었다.

소프트웨어의 경비를 이상 열거한 바와 같은 것들을 모두 포함시킨다면 — 직원들에게 새로운 소프트웨어를 쓸수있게 시키는 재교육과, 소프트웨어의 개발, 기존 소프트웨어의 유지 수정등에 드는 제비용 — 어느 기관이나 시간이 지날 수록 하드웨어를 바꿈으로써 소요되는 소프트웨어 경비는 눈덩이처럼 늘어나게 마련이다. 전산요원의 수가 늘어나므로 교육비가 더 들고, 운용중이던 소프트웨어가 시간이 지날수록 늘어나므로 그것을 유지, 보수, 수정하는데 막대한 재정적 부담을 감수해야 한다.

이와같은 문제는 몇가지 방법으로 해결될 수가 있다. 하드웨어가 새로운 것이 나오더라도 기존 소프트웨어를 그대로 수정시키지 않아도 되게 만들 수가 있다. (소위 IBM, VAX, PDP, 3B등에서 보는 "family of computer" 개념이다) 회사의 업무량이 늘어날수록 처음에는 소형 컴퓨터로부터 점차 대형 컴퓨터로 대체시켜가되, 새기종을 바로 이 family에서 고른다면 소프트웨어의 수정이나 직원의 재교육 문제가 비교적 감소될 수가 있다. 그러나 이와같은 방식은 일단 처음에 어느 기종의 컴퓨터를 택하고 나면 영원히 그 family of computer에 머물러야 한다는 단점이 있다. 경우에 따라서는 그 family에 속한 컴퓨터의 종류가 제한적이고 부적합해질 경우도 있고, 또는 회사의 업무 양상이 바뀔때마다 전혀 다른 새기종으로 기계를 바꾸어야 할 필요를 느낄때도 있다. 이와같은 문제는 OS로 해결할 수가 있다. OS로 이러한 문제를 해결하는 방법중 하

나는 소위 OS의 이식성(portability)을 높게 만드는 것이다. "이식성"이란 것은 한 소프트웨어를 그것이 수행하던 하드웨어로부터 다른 하드웨어로 옮겨 수행할 수 있도록 만드는 데 드는 시간과 노력이 매우 적다 것을 뜻한다. 만일 OS가 이식성이 좋으면 어떤 새로운 하드웨어가 나오더라도 종래의 OS를 그 새로운 하드웨어에 이식시키어 버리면 최종 사용자는 그대로 예전 OS를 사용할 수 있고 따라서 재교육이나 기존 소프트웨어의 수정 문제도 발생하지 않는다. UNIX나 CP/M은 이러한 이식성이 매우 좋은 OS로 평정이 있고 매우 다양한 하드웨어에 이식이 되어 돌아가고 있다. 일단 UNIX나 CP/M으로 공부를 하고 프로그램을 개발하고 나면 어떠한 하드웨어로 옮기더라도 최종사용자가 재교육을 받거나 기존 소프트웨어를 수정해야 할 필요가 적어진다. 미래의 OS는 갈수록 이 이식성을 좋도록 만들것이다. 이식성이 좋은 OS는 여러 가지 다양한 하드웨어에 돌아가므로 그러한 OS를 위해 응용프로그램을 작성하면 훨씬 수익성이 높아지게 되고, 따라서 이식성이 낮은 다른 OS보다 단시일내에 방대한 양의 응용프로그램을 갖추게 된다. 또한 이식성이 높은 OS는 여러기종의 하드웨어에 운용되므로 서로 다른 하드웨어라도 서로 연결하여 네트워크를 형성하기도 쉽게 된다.

이 이식성과는 전혀다른 방식으로 상기 문제를 해결한 것이 IBM의 VM이라는 OS이다. VM은 과거 IBM에서 운용되던 어떤 OS도 다 그대로 한 하드웨어에

서 사용할 수 있게끔, 여러 OS들을 동시 운용 함으로써 최종 사용자가 새 OS를 배워야하는 문제를 해결 시키어주는 방법을 쓰고 있다.

결론적으로 미래의 OS는 전문적이고 소형의 OS가 될 것이다. 하드웨어의 값이 갈수록 싸지는 반면 소프트웨어의 가격은 갈수록 올라가는 추세이므로 미래의 OS는 최종사용자의 생산성을 향상시키는 것이 하드웨어를 효율적으로 운용하는 것보다 더욱 중요한 문제가 될것이다. 입출력을 더 편리하게 만들 것이고, 이식성을 높이어 한번 한 OS의 사용법을 익히면 어느 하드웨어로 바뀌더라도 최종 사용자에게는 영향이 안미치도록 할 것이다. 네트워크와 분산처리개념은 미래 OS에서 더욱 확산될 것이다. 하드웨어가 싸지고 소형화할수록 병렬처리 개념이 더욱 확산될 것이고 이 병렬처리.문제를 효율적으로 처리하는것이 OS의 큰 과제 중의 하나가 될 것이다. 네트워크와 분산처리가 많이 쓰일수록 보안문제와 신뢰도 문제가 크게 대두될 것이다.

이상과 같은 미래 OS를 실현시키기 위해서는 여러 가지 많은 난제들이 가로놓여 있다. 분산 화일문제, deadlock 문제, synchronization문제, 프로세스간 통신 기법, 병렬처리의 제어문제, 신뢰성 문제등이 그 예이다. 그러나 필요는 발명의 어머니라는 말이 있듯이 초 집적회로기술의 혁명은 OS에도 새로운 혁명을 조용히 그러나 확실히 수행시키어 나갈것이다. *

♣ 안 내 ♣

국제학술회의 계획

■ 고도정보화 기술에 관한 STOP OVER 세미나 개최

일 시 : 1985년 9월 12일~9월 14일

■ 한일 합동 전자통신 학술회의

일 시 : 1985년 11월 중

- 내 용 : 1. 한일 자동번역 시스템
2. 차세대 컴퓨터 시스템
3. 종합정보 통신망