

UNIX 개발, Linux activity

고려대학교 유 혁*

유닉스는 윈도우(Windows)와 함께 20세기에 가장 중요한 위치를 차지하는 운영체제이다. 60년대 후반에 개발되었지만, 유닉스에서 사용되었던 주요 추상개념(abstraction)들은 바뀌지 않고 30년이 지난 지금에도 사용되고 있으며, 마이크로소프트사의 NT 운영체제를 포함하여 많은 운영체제에 직 간접적으로 영향을 미쳤다. 전 세계의 수많은 학교와 연구소에서는 아직도 유닉스에 기반 하여 연구와 개발이 이루어지고 있다.

유닉스는 미국 벨연구소의 Ken Thompson이라는 개인에 의하여 개발되었다. 벨연구소의 구석에 아무도 쓰지 않던 PDP-7을 발견한 Thompson은 개인의 호기심으로 개발을 시작하였고 Dennis Ritchie라는 사람이 C언어로 다시 유닉스를 쓰게 된다. 당시에는 운영체제는 어셈블리(assembly)언어로 쓰여야 한다는 관념이 팽배해 있었는데, 고급언어인 C언어로 운영체제를 쓴다는 것은 매우 위험한 시도였다. 그러나 C언어로 쓰여진 유닉스는 소위 이식성(portability)을 제공하게 되어 엄청난 성공의 발판을 구축하게 된다.

유닉스라는 이름은 멀틱스(Multics)라는 이름에서 나온 것으로, 멀틱스는 '60년대 초반에 미국의 MIT대학에서 개발하던 대규모 운영체제로서, 소문나게(?) 실패한 프로젝트이다. 유닉스의 저자들은 멀틱스 프로젝트에 참여했던 기억으로부터 유닉스라는 이름을 붙이게 되었다. 실제로 유닉스에서 사용한 여러 추상개념(예를 들면 파일 시스템의 구조)들이 멀틱스에서 가져온 것이다. 비

록 멀틱스는 실패했지만 그때의 경험이 축적되어 있었기 때문에 유닉스가 제대로 될 수 있었다고도 할 수 있다.

현재의 관점에서 생각하면 혼자서 운영체제를 쓰는 것이 가능하라는 의문도 생기지만, 그때 당시만 하더라도 메모리와 프로세서 같은 하드웨어가 매우 간단하고 입출력도 극히 초보적인 형태였기 때문에 운영체제에 가장 핵심적인 커널 자체의 크기는 수K바이트에 불과했다. 따라서 당시 운영체제를 쓰는데 있어서 어려운 점은 코드가 복잡한 것이 아니라 운영체제의 구조(architecture)가 제대로 정의되어 있지 못했다는 것이다. 유닉스가 운영체제 발전에 주요하게 기여한 점은 커널의 구조의 기초를 세웠다는 데 있다. 이후에 커널이 복잡하게 되지만, 이러한 복잡성은 운영체제의 구조가 복잡해지는 것이 아니라 다양한 하드웨어를 지원하고 다 수의 사용자를 지원하기 위하여 발생하는 복잡성이 된다.

유닉스의 구조는 매우 단순하면서도 효율적으로 되어 있어서, 짧은 시간 안에 많은 사용자를 확보하게 되었다. 벨연구소 외부에서 사용된 유닉스는 Version 6로서 1976년에 발표되었으며, 1978년에는 연구용이 아닌 소프트웨어 제품으로서 등장하게 되었다. 그후에 개량을 거듭하여 1983년에 System V가 AT&T(벨연구소의 모회사)를 통하여 널리 보급되게 되었다.

유닉스가 중요하게 기여한 다른 하나는 네트워크부이다. 이것은 BSD(Berkeley Software Distribution)을 통해서인데, BSD란 UC Berkeley에서 만들어서 무료로 학교와 연구소에 보급한 유닉스를 일컫는다. BSD는 숫자로 버전

* 증신희원

을 나타내는데, 그 중에서 1983년에 발표된 4.2 BSD는 네트워크 프로그래밍을 기본 인터페이스인 소켓(socket)을 도입하였고, TCP/IP 프로토콜을 매우 효율적으로 구현하였다. 이것은 현재 인터넷에서 TCP/IP가 주요 프로토콜로서 자리매김을 하는데 결정적인 역할을 하였다.

BSD 유닉스는 학교와 연구소뿐 아니라 회사에서도 채택되어 IBM의 AIX, Digital사의 Ultrix 등 상용 운영체제의 기반이 되었고 AT&T의 System V보다 더 널리 사용되게 되었다. 이처럼 벨연구소에서 시작된 유닉스는 버클리 대학을 통하여 꽃을 피우게 된다. 주목할 점은 BSD의 개발의 주역들이 박사과정의 학생들이었다는 것이다. 물론 이들은 학교만 다닌 학생들이 아니라 산업체에서 일하면서 경험을 쌓고 난 후에 박사과정에 진학하여 진가를 발휘한다. 이러한 학생을 입학시키고 키운 버클리 대학이 유닉스를 세계적인 시스템으로 만들었다고 하겠다.

4.4BSD를 마지막으로 유닉스의 개발은 멈추게 된다. 개발이 멈춘 주요 이유 중의 하나는 유닉스에 대한 저작권을 가진 AT&T와 BSD그룹간의 충돌이었다. 이후 BSD그룹은 사실상 해체되고, 컴퓨터 회사들이 유닉스를 탑재한 워크스테이션을 선보이게 되었다. 상용 유닉스를 만들 때에 각 회사들은 차별화를 위하여 교유의 기능을 추가하였다. 이것은 유닉스라는 이름은 같지만 컴퓨터 회사에 따라 응용프로그램이 제대로 동작하는 않은 호환성의 문제를 야기하여 사용자들의 불편을 초래하였다. 또한 유닉스는 워크스테이션 같은 고성능의 컴퓨터 시스템에서만 동작을 하였고 PC에서는 마이크로소프트의 윈도우 시스템이 견고하게 자리를 잡았다.

BSD 이후의 공백을 메우기 시작한 것이 리눅스(Linux)이다. 유닉스의 시작과 비슷하게, 리눅스도 한 개인의 취미에서 시작했다. 헬싱키대학의 Linus Torvalds가 1991년경에 당시 교육용으로 사용되던 Minix라는 운영체제를 개선하기 위하여 만들었다. 유닉스는 이미 너무 크기가 커져서 교육용으로 사용하기에는 부적합했고, Minix는 교육용으로 PC에서 동작하도록 만들어진 조그만 유닉스였다. 그러나 Minix는 개발도구가 부족하고 사용하기에 매우 불편했다. 리눅스

는 Free Software Foundation의 GNU 컴파일러 같은 도구들을 이용하여 편리한 개발환경을 제공하여 급속한 발전을 하게 된다.

리눅스의 발전에 가장 중요한 역할을 한 것은 리눅스 초기부터 소스코드를 무료로 개방하고 커널에 대한 지식이 있는 사람은 누구든지 개발에 참여할 수 있도록 한 것이다. 즉, 초기 유닉스와 BSD가 하나의 그룹을 중심으로 개발이 이루어졌지만, 리눅스는 전세계에 퍼져 있는 능력 있는 많은 사람들이 리눅스에 새로운 기능을 추가하고 성능을 향상시키는데 기여한 것이다. 이것은 리눅스를 단기간에 안정적으로 동작하게 만들고 필요한 기능을 갖춘 운영체제가 되도록 하였다. 운영체제 코드는 매우 난해하고 복잡하여 고도로 훈련된 사람도 실수를 하기 쉽다. 이런 실수를 줄이는 방법은 코드를 다른 사람들이 검토를 하여 잘못된 부분을 고치도록 하는 것이다. 즉, 개발자간에 코드리뷰(code review)를 통하여 버그를 잡아내는 것인데, 이 작업은 많은 시간이 소요되며 개발자간에 충돌이 생기기도 한다. 수천 명이 커널 개발에 참여하는 윈도우NT의 경우 아직도 문제가 속출하고 있다는 것을 볼 때 하나의 새로운 운영체제를 안정화시키는데 긴 시간이 소요됨을 보여준다. 리눅스는 개발된 코드가 전세계에 자발적으로 참여하는 개발자들에 의하여 검토되기 때문에 이상적인 코드리뷰가 이루어진 다. 이점에서 리눅스는 개방소스코드(open source code)라는 커널 개발의 새로운 패러다임을 제시한 것이라 하겠다.

리눅스의 또 하나의 강점은 커널이 하나의 버전으로 통일되어 호환성의 문제를 해결한다는 것이다. BSD에 기반 하는 상용 운영체제들의 경우 조금씩 다른 커널을 사용해서 호환성 문제가 발생했는데, 리눅스는 개발자는 많지만 하나의 커널만이 배포되어 사용되고 있어서 사용자들이 과거 유닉스에서 겪었던 호환성의 어려움을 당하지 않는다. 또한 리눅스는 스타오피스와 같이 PC에서 사용되는 응용프로그램들이 존재하여 개발자가 아닌 일반 사용자들에게도 매력적이다. 전세계에서 다양한 사람들이 개발에 참여한다는 것이 응용프로그램의 개발도 촉진시키고 있다.

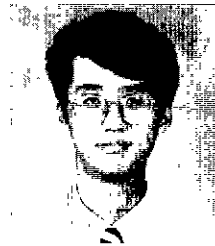
리눅스의 미래에 대하여 전문가들의 의견은 엇갈린다. 그러나 확실한 것은 과거 2, 3년전보다

리눅스의 밝은 미래를 점치는 사람이 훨씬 늘어났다는 것이다. 특히 최근에 기업을 공개한 Red Hat의 주식이 \$100을 넘어가고 실리콘 벨리에서 리눅스의 열풍이 불고 있는 것을 감안할 때 리눅스가 과거 유닉스가 학교와 연구소에서 머물던 것을 넘어서리라 생각된다. 또한 마이크로소프트가 독점에 관련된 재판의 1심에서 불리한 판결을 받은 현 시점에서 볼 때 리눅스의 가능성의 더욱 올라가고 있다.

국내에서의 운영체제의 연구 그리고 개발이라는 면에서 볼 때 리눅스는 새로운 가능성을 보여준다고 하겠다. 그 동안 국내에서 몇몇 학교와 연구소를 중심으로 운영체제에 대한 개발 노력이 있었으나, 개발환경의 문제, 응용프로그램의 부재 등으로 인하여 제대로 이루어지지 못했다고 해도 과언이 아니다. 그러나, 리눅스는 국적도 없고 저작권도 없기 때문에 기술력의 여부에 따라 또한 효율적인 투자를 통하여 우리 나라도 운영체제의

강국이 될 수 있는 것이다. 새로운 천년을 리눅스에 대한 기대와 국내 운영체제분야의 여러 선후배 여러분의 분발에 대한 기대로 시작하면 좋겠다.

유 혁



1982 서울대 전자공학과 학사
 1990 미시건 대학 전산학 학사
 1990~1995 Sun Microsystems 연구원
 1995~현재 고려대학교 부교수
 관심분야 운영체제, 네트워크, 멀티미디어 스트리밍
 E mail hxy@joy.korea.ac.kr

'99 정례회의 및 편집위원회 연간일정표

월 별	정 령 회 의		편 집 위 원 회	
	상임이사회	정례이사회	논 문 지	학 회 지
1월	8일(금) 17:00		29일(금) 16:00, 전체회의	20일(금) 16:30
2월	5일(금) 16:00	26일(금) 18:00	26일(금) 16:00	19일(금) 16:30
3월	5일(금) 16:00		26일(금) 16:00, 전체회의	19일(금) 16:30
4월	9일(금) 16:00	16일(금) 17:00	30일(금) 16:00	23일(금) 19:00
5월	7일(금) 16:00		28일(금) 16:00, 전체회의	21일(금) 16:30
6월	11일(금) 16:00	25일(금) 17:00		18일(금) 16:30
7월	9일(금) 16:00		2일(금) 16:00	16일(금) 16:30
8월			27일(금) 16:00, 전체회의	20일(금) 16:30
9월	3일(금) 16:00	10일(금) 17:00		17일(금) 16:30
10월	4일(월) 16:00	15일(금) 17:00	29일(금) 16:00, 전체회의	22일(금) 19:00
11월	5일(금) 16:00		19일(금) 16:00	19일(금) 16:30
12월	1일(수) 16:00	17일(금) 17:00	17일(금) 17:00, 전체회의	17일(금) 15:00

※ 회의일정은 사정에 따라 변경될 수 있음.