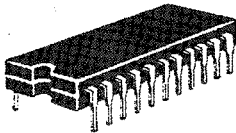


吳 吉 祿

韓國電子通信研究所
컴퓨터연구부장/工博

UNIX Operating System의 발전 동향 (I)



한국전자통신연구소가 발행하는 전자통신 동향 분석(제 1 권 제 1 호)에 컴퓨터 개발부 시스템 소프트웨어 연구실 임기욱실장께서 작성한 것으로 2 회에 걸쳐 연재한다.

UNIX O. S. (Operating System)에 관한 모든 것을 지면상으로 한꺼번에 설명하기는 쉽지 않다. 그 이유는 UNIX의 종류가 무척 많을 뿐만 아니라 그 기술 발전속도가 너무 빠르기 때문이기도 하다. 이러한 상황을 감안하여 본고에서는 UNIX O. S의 현황, 발전추세, UNIX를 기본으로 하는 컴퓨터 시스템 등으로 크게 나누어 UNIX O. S. 에 관련된 발전동향을 살펴 보고자 한다.

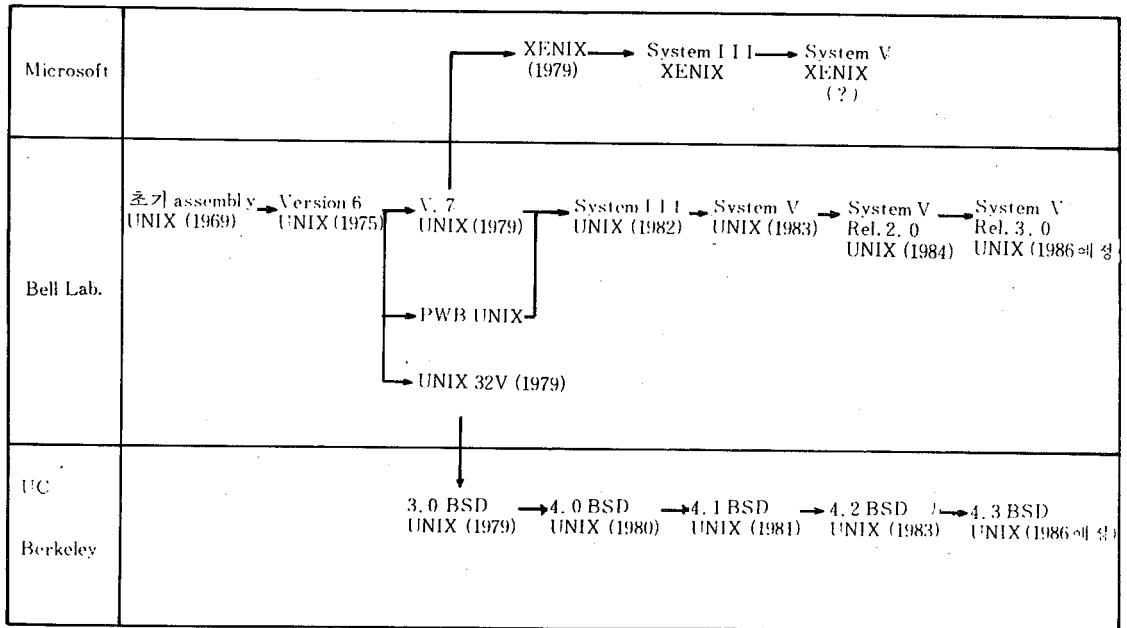
I. UNIX Operating System의 현황

현재 세계적인 표준 운영체제가 되어가고 있는 UNIX O. S. 는 1969년 Bell Lab. 의 Ken Thompson이 어셈블리 언어로 PDP 7 용 UNIX 를 작성하면서 시작되었다. 이 UNIX O. S. 는 1973년 C언어로 재작성되었으며, 1977년도까지는 DEC사의 PDP 11 family용 UNIX만이 개발되었으나 Johnson과 Ritchie에 의해서 machine dependent한 부분이 연구되어 최초로 PDP family가 아닌 interdata computer system에 UNIX 가 이식되면서부터 여러가지 다른 기종의 컴퓨터에 이식되어 여러가지 version의 UNIX가 탄생하기에 이르렀다. <(그림 1) UNIX Lineages 참조>

본고에서는 이러한 여러가지 UNIX version 중에서 현재 가장 큰 주류를 이루고 있는 System V UNIX와 4.2BSD(Berkeley Software Distribution) UNIX를 비교 분석해 보고 현재 16비트 마이크로 프로세서를 기본으로 하는 시스템에서 가장 많이 사용되고 있는 XENIX O. S. 에 대하여서도 간단히 설명하기로 한다.

1. System V versus 4.2BSD

AT&T는 1983년 초 UNIX System V를 발표



(그림 1) UNIX Lineages

하였으며, 1984년 초에는 System V release 2.0를 발표하였다. AT&T는 UNIX System V release 2.0를 발표함으로써 상업적 사용자들의 관심을 끌고 있다. 이는 AT&T가 UNIX System V를 공식적으로 지원하기 시작함으로써 R&D 영역에서 벗어나 본격적인 상업적 영역으로의 진출을 의미하는 것이기도 하다.

AT&T사의 UNIX System V에 대한 기본 전략은 마이크로부터 메인 프레임 컴퓨터 전반에 걸쳐 UNIX System V를 주축으로 하는 새로운 소프트웨어 environment를 만들겠다는 것이다. 즉, 가능한 한 넓고 다양한 종류의 하드웨어에 걸쳐 UNIX System V가 porting되게 하기 위하여 upward호환성을 유지하면서, UNIX 시스템의 구조와 원시코드에 대하여 계속 개방 정책을 취하겠다는 것이다.

현재까지 알려진 System V의 장점으로는 간결성과 호환성이 뛰어나다는 것이다. 간결성(Simplicity)은 처음 UNIX가 탄생하였을 때의 기본 목표의 하나로서 필요없는 부분을 과감하게 삭제 또는 포함시키지 않음으로써 성능 향상에 기여하자는 것이며, 호환성(Compatibility)은 AT&T의 전략에 따른 것으로 원시코드 및

목적코드에 대한 호환성 모두를 유지하고 있다. 또한 이 시스템은 kernel을 공유할 수 있도록 하여 다중처리 시스템을 지원할 수 있으며 기존의 512바이트 블록의 파일 시스템을 512 바이트 또는 1,024바이트 블록 파일 시스템으로 고쳤으며, 파일 및 레코드에 대한 locking mechanism, 가상 터미널 기능 등을 제공하고 있다.

이에 비하여 1984년에 발표된 4.2BSD의 기본 개념은 필요한 기능을 모두 찾아내어 한꺼번에 모두를 재정리함으로써 새로운 basis를 만들고자 하는데 있다. UNIX 4.2BSD O. S.은 4.1BSD O. S.를 완전히 새로 작성한 것으로 kernel에 LAN기능을 강화시켰으며 4K 바이트 블록의 고속파일 시스템을 제공하고 있다. 4.2 BSD는 AT&T의 UNIX에 비하여 demand-paged 가상 메모리 기능(AT&T UNIX는 System V Rel. 2.0 이상에서 제공) 및 full screen editor인 vi, termcap, curses, pascal, lisp, sdb 등 약 80개의 새로운 command들을 제공하고 있다.

4.2BSD UNIX의 장점으로는 remote login, remote exec, remote print, remote copy, ftp 등의 server 제공, server들을 위한 socket, re-

ndezvous, layers of protocols 등의 interprocess communication mechanism 제공과 foreground/background의 전환이 용이하여 job control이 편리한 점 등이 있다.

또한 System V release 2.0을 자세히 살펴 보면, Berkeley-style의 enhancement를 취했음을 알 수 있는데 vi editor나 pg('more'의 다른 이름), job control을 제공하는 새로운 형태의 shell, purdue 대학의 terminfo에 근거한 curses package, C shell의 aliases를 일반화한 shell function들을 찾아볼 수 있으며, demand-paged 가상 메모리에 대한 시장수요에 따라 이를 구현하기로 했다. 즉, 연구환경에서의 우수한 제품을 시장요구에 따라 민감하게 도입하고 있다. 그럼에도 불구하고, System V는 네트워크 분야에 취약점을 가지고 있다.

반면에 4.2BSD는 TCP/IP 및 다른 프로토콜들을 지원함으로써 네트워크 분야를 강조하고 있다. 이에 따라 System V 설계자들은 "Ethernetcompatible"한 3B-net을 발표하여 AT&T PC사용자들로 하여금 파일을 공유하게 하고 파일 server처럼 동작하는 central AT&T Computer와 데이터를 교환할 수 있게 하였다.

이러한 PC용 3B 소프트웨어 인터페이스는 직접 System V 내에 쓰여졌으며 사용자로 하여금 central UNIX machine 상에서의 터미널 emulation과 native PC O. S. 인 DOS 사이에서 정보를 교환할 수 있게 하였다. 그리고 한 개 이상의 DOS directory를 central UNIX machine에 physical하게 상주시킬 수 있게 함으로써 DOS 환경하에서 동작하는 프로그램들이 UNIX machine에 resident한 파일을 사용할 수 있게 하였다. 그러나 이러한 것도 네트워크로 연결된 4.2BSD based machine에서 remote login하여 파일을 전송하는 것에 비하면 훨씬 못 미치는 것이다. 그러나 이러한 사실만으로 System V 또는 4.2BSD를 선택하는 것은 곤란하다.

일반적으로 UNIX O. S. 를 선정하는 데는 다음과 같은 점들을 고려하는 것이 좋을 것이다.

- 앞으로의 전망 : End user의 code licence

및 binary licence, software house들 및 컴퓨터 시스템 제조업자의 경향

- 기능 : Memory management 기능, networking 기능, device제공 기능, job control 기능 및 inter-process 통신 방법

- 신뢰성 : Bug의 갯수 및 심각성, 시스템 가용성

- 호환성 : 기존 프로그램, 새로운 프로그램 및 third party 프로그램과의 호환성

- 성능 : 속도, 메모리 및 디스크의 용량, 가격 대 성능비

상기 사항을 고려하여 System V와 4.2BSD를 비교하여 보면 System V가 전반적인 간결성(4.2BSD의 61938 원시코드 라인에 비하여 System V는 29,000 라인)에 의한 신뢰성 및 성능 호환성이 좋은 반면 4.2BSD는 System V에 비해 여러 가지 좋은 기능을 지니고 있음을 알 수 있다. 그러나 System V는 AT&T 자체연구와 대학에서 개발된 좋은 기능을 선별하여 계속 추가할 것이며, 4.2BSD UNIX는 DEC사, Sun microsystem사, National Semiconduction사와 같은 전문업체로부터의 혁신적인 제품을 기대할 수도 있다.

지금까지의 경향을 살펴보면 마이크로 또는 소형 미니 컴퓨터 시스템에는 System V, 미니 및 슈퍼 미니 이상의 컴퓨터 시스템에는 4.2BSD가 주로 사용되고 있으며, 상업용으로는 System V가 연구소 및 학계에서는 4.2BSD가 선호되고 있다는 느낌이다.

2. XENIX O. S

1980년 중반 Microsoft사는 새로운 시대의 16비트 마이크로 프로세서를 위한 강력한 operating system을 제공하기 위하여 UNIX시분할 시스템의 한 version인 XENIX O. S. 을 개발하기 시작하였다. 1983년 이래 Microsoft사는 자사의 XENIX O. S. 을 UNIX시스템을 위한 바이너리 프로그램의 표준으로 정착시키기 위하여 노력을 경주하여 왔다.

바이너리 프로그램을 표준화시킴으로써 Mic-

Microsoft사는 소프트웨어 개발업자들로 하여금 어떤 한 version의 응용 소프트웨어가 개발되면 그 소프트웨어를 여러 종류의 컴퓨터에서 동작할 수 있게 하여 유지보수가 가능한 UNIX환경을 조성함으로써 XENIX의 판매를 촉진하고자 했다. 이러한 노력의 결과로 XENIX는 Intel286 마이크로 프로세서에 개발되어 오늘날 IBMPC, Tandy 등을 포함한 주요 마이크로 컴퓨터 시스템에 사용되어 10만개 이상의 카피가 팔려왔다. 이러한 기반은 서로 다른 마이크로 컴퓨터들 사이에 호환성있는 설비를 제공하게 함으로써 소프트웨어 이식에 대한 비용을 현격히 줄일 수 있으므로, 더욱 더 판매에 유리하게 되었다.

현재 Microsoft사는 내년 하반기까지 약 40만개의 XENIX 카피가 팔릴 것으로 예측하고 있다. 이러한 근거로는 IBM PC-AT에 복수 사용자 O. S. 으로 쓰일 것이라는 점, 소프트웨어 개발업자들이 XENIX의 바이너리 프로그램 표준에 의한 이점을 생각하여 소프트웨어 개발환경으로 XENIX를 사용할 것이라는 확신, XENIX를 AT&T의 System V화 하겠다는 Microsoft사의 계약 등이 XENIX 판매에 도움을 주리라는 전망을 바탕으로 하고 있다.

Microsoft사는 AT&T사의 System V 인터페이스 정의에 의한 XENIX의 System V화를 이룩함으로써 System V와 MS-DOS와의 호환성을 유지하려 하고 있다. 즉, XENIX System V는 코스볼, 파스칼, 포트란, C, 베이직 인터프리터 등 마이크로 소프트웨어의 표준언어를 MS-DOS와 호환성있게 사용할 수 있고, IBMPC-AT가 지원할 여러 종류의 하드웨어 옵션이 가능하도록 loadable device drive를 제공할 것이며, XENIX System V와 MS-DOS 사이의 transparent한 네트워크를 제공할 계획으로 있다. 이러한 XENIX System V는 UNIX System V가 원시코드 portability 특성을 갖는데 반해 강력한 바이너리 프로그램 표준 특성을 갖게 될 것이므로 더욱 더 응용 소프트웨어 개발업자들에게 흥미를 끌 것으로 생각된다.

II. UNIX Operating System의 발전 추세

현재 UNIX O. S. 의 발전 추세중 주목할 만한 것으로는 AT&T의 System V rel. 3.0의 발표와 Berkeley의 4.3BSD의 발표, System V rel. 3.0에 들어 있는 RFS(remote file sharing)과 4.2BSD 상에서 개발된 NFS(network file system) 등의 분산 파일시스템, UNIX의 국제화 일환으로 개발된 Japanese UNIX 등이 있다.

1. Changes to the Kernel in 4.3 BSD

4.2BSD에는 System V rel. 1.0.에서는 제공되지 않던 우수한 feature인 TCP/IP 기능과 가상 메모리를 제공하는 기능 등 여러 다양한 기능을 제공함으로써 사용자를 많이 확보하였으나, 점차 네트워크에 관계된 성능과 O. S. 의 보다 우수한 성능의 요구가 커짐에 따라 4.2BSD의 개선은 필연적이었다.

4.3BSD는 이러한 시대적 요구에 부응하여 출현하게 되었으며, 나아가 4.4BSD 혹은 4.5BSD에서는 분산 파일 시스템을 지원하는 기능을 갖게 될 것이다. 4.2BSD와 비교해서 4.3BSD에서의 가장 크게 변화된 점을 분류하면 다음과 같다.

- bug fixes 및 performance improvements
- 새로운 하드웨어와 소프트웨어 시스템을 수용하기 위한, 혹은 하드웨어나 프로토콜을 특정지우는 코드를 제거하기 위한 골격의 일반화
- 새로운 프로토콜과 하드웨어 지원
- 화일시스템 이름을 번역하는 오버헤드를 감소하기 위한 캐쉬의 사용
- namei의 name lookup 기능에 새로운 인터페이스를 마련하여 argument들의 encapsulation과 namei routine call의 정보와 side effect를 제공
- 네트워크의 공통부분들로부터 대부분의 internet dependency들의 제거 및 같은 네트워크 하드웨어 상의 multiple address family들의 보다 많은 허용

- XNS 네트워크 프로토콜을 위한 지원
- VAX 8600, 8650 프로세서 (UNIBUS와 MASSBUS peripheral들을 가지고 있음) 들을 지원
- 새로운 driver들의 개발 (for DHU11, DMZ 32 terminal multiplexers, the TU81 and other TMSCP tape drives, the VS 100 display, the DEUNA, Excelan 204, and Interlan NP 100 ethernet interfaces, and the ACC HDH and DDN X. 25IMP interfaces)
- 64K와 256K 메모리 칩들을 사용하는 VAX 11/780과 11/785상의 MS780-E메모리 제어기를 위한 완전 지원

2. AT&T의 RFS와 SUN의 NFS

VLSI기술의 발전으로 컴퓨터 H/W는 급속히 발달하여 computing power 및 메모리 용량이 훨씬 값싸게 제공되고 있으며, 앞으로도 이런 추세는 계속 진행될 전망이다. 이런 현상은 컴퓨터 시스템의 개발에 많은 영향을 주어 더 많은 프로세서 혹은 더 많은 컴퓨터 시스템들을 연결시키기 위한 병렬처리 시스템과 구내정보 통신망에 기반을 둔 분산시스템에 대한 연구가 활발히 진행되게 한다. 이러한 연구의 주 목표는 여러 computing power를 합쳐서 높은 performance 시스템을 이루기 위한 것과 컴퓨터 시스템 사이에 원활한 정보자료의 분배를 제공하는 데 있다.

Distributed 파일 시스템의 기능을 첨가할 때는 통신 소프트웨어를 UNIX의 하부 구조에 integration시키는 일과, 그 위에서 수행되고 있는 UNIX 파일 시스템이 네트워크 상에 넓게 연결될 수 있도록 UNIX 파일 시스템을 수정하는 작업이 필요하다. 이때 지켜져야 할 기본적인 원칙은 UNIX 파일 시스템의 kernel이 수정된다고 하더라도 사용자에게는 기존의 파일 시스템을 사용하는 것과 같은 파일 시스템의 호환성을 제공해야 한다는 점이다.

또한 사용자는 네트워크를 통해 원격 데이터

를 액세스하고 있는 동안이라도 사용자 측면에서는 마치 한 시스템을 사용하고 있는 것처럼 보이게 하기 위하여 transparency, 신뢰성, concurrency control 및 UNIX 파일 시스템의 semantics, 데이터와 computing resource의 보안성을 유지하여야 한다.

현재 UNIX를 기본으로 하여 개발된 분산 시스템으로는 LOCUS (UCLA), Newcastle Connection (Newcastle University), IBIS (Purdue University), AEGIS (APPOLLO), NFS (SUN), RFS (AT&T) 등이 대표적이다. 이들은 모두 각 시스템마다 UNIX 이미지를 갖고 자율적으로 수행되는 NOS (Network Operating System) 들이다.

Distributed UNIX 파일 시스템을 설계할 때는 다음과 같은 요소가 고려되어야 한다.

- 네트워크 Wide 한 single naming tree 의 구성
- 통신용 소프트웨어 및 하드웨어
- 사용자에게 제공되는 원격 시스템 액세스의 transparency
- 구현이 일어나는 위치
- Homogeneous 시스템 혹은 heterogeneous 시스템

위의 열거한 요소들은 독자적인 특성보다는 서로 밀접한 연관을 갖고 있으므로 한 가지 요소에 대한 결정은 다른 요소에도 그 영향을 주게 된다.

가. AT&T의 RFS (Remote File System) RFS의 설계 목표는 다음과 같다.

- 원격 파일 시스템을 로컬 시스템에 remote mount시켜 마치 로컬 시스템내에 상주하는 파일 시스템을 사용하듯이 하는 environment 를 제공하는 transparency.
- 로컬 시스템 관리자에게 데이터의 보안과 integrity를 제공하는 보안 메카니즘
- Unix파일 시스템의 semantics의 유지
- 동시 파일 액세스

RFS의 주된 장점을 unix 파일 시스템의 semantics를 유지한다는 점이다. 이는 remote와 local이 정확하게 같은 방법으로 동작한다는 것

이다. 예를 들면 server는 얼마나 많은 파일이
 오픈되었는지를 알고 있어서 unlink 후 파일이
 삭제될 때를 결정할 수 있다는 것이다. 만약
 server의 이러한 기능이 없다면 다른 사용자가
 어떤 한 파일을 사용하고 있는 동안 그 파일이
 삭제될 위험이 있다. 파일의 일관성을 위해서는
 client machine에서 그 파일을 caching하지않고
 server에 의해 관리되는 하나의 version만을 이
 용한다. 이는 모든 I/O operation이 네트워크를
 통하여 해당 server에 의해서만 통제되는 것을
 의미한다. 이럴 경우 네트워크가 로컬 disk drive
 에 비해 너무 느리거나 혹은 server의 부하가 매
 우 심할 때는 큰 문제가 될 수 있다.

SUN의 NFS는 이러한 문제를 파일 블럭 caching에 의해 해결하고 있다. 즉, remote read시에
 파일의 일부분을 caching하여 read하고 remote
 write시에는 원격장치에 직접 write한다. 이러
 한 방법은 네트워크 성능을 개선하지만 incons-
 istent 데이터를 관리해야 하는 부담을 안고 있
 다. SUN의 NFS와는 달리 RFS는 stateful server
 를 가지고 있어 사용자들에게 네트워크를 통한
 원격장치를 액세스 가능하게 한다. 또한 NFS와
 는 달리 데이터 integrity를 위한 concurrency 조
 절이 가능하다.

RFS에서의 가장 큰 약점은 name서비스이다.
 만약 한 machine이 꺼지면 네트워크의 어떠한
 machine들도 그들의 파일 시스템을 remote machine
 들에게 알리거나 원격 파일 시스템을 mount할
 수 없다. 이러한 문제를 해결하기 위해서 시스
 템 관리자는 한 machine의 고장시 그 machine의
 기능을 양도받을 machine을 지정한다. Name 서
 비스의 고장은 기존 mount된 파일 시스템에는 영
 향을 미치지 않고 더 이상의 mount를 할 수 없
 게 한다.

네트워크 상의 machine사이의 보안은 원격사
 용자 ID, 그룹 ID와 로컬사용자 ID, 그룹 ID와
 mapping을 통해 보안을 유지한다.

나. SUN의 NFS

NFS의 설계목표는 다음과 같다.

- RFS와 같은 transparent 파일 액세스 제공

- 네트워크의 불완전 시에 신뢰도 제공
- Unix 파일 시스템의 semantics의 유지
- UNIX가 아닌 O.S. 도 지원

NFS의 네트워크 파일 시스템의 접근방식은
 여러 recovery, 시스템 independence 그리고 가용
 도를 포함한 여러 장점들을 제공한다. NFS는 예
 러 recovery를 어렵게 만드는 status에 관한 정
 보를 빠르고 쉽게 제거함으로써 예러 recovery
 를 하며, RFS와는 달리 고장시 어떠한 status에
 관한 정보도 저장하지 않기 때문에 RFS에서 처
 럼 간헐적으로 네트워크 고장에 대해 공고해야
 하는 부담이 없다.

NFS는 standard vendor와 분산 파일 시스템에
 독립적인 O.S를 정의함으로써 사용자들에게 여
 러 다른 기종들 사이에서의 데이터 분할을 제공
 한다. 하지만, 훌륭한 standard가 되기 위해서
 는 산업표준위원회와 IBM, DEC, AT&T 등 주요
 vender들의 도움이 요구된다. NFS의 가장 큰 장
 점은 가용도로서 SUN, Pyramid, Gould, Celerity,
 Sequent에 의해 협조되고 있으며 Mt.Xinu와 Lach-
 man협회도 Vax system에서의 NFS를 협조하
 기로 공표한 바 있다.

RFS와 NFS의 공통 약점은 default로 파일 분
 할이 되지 않음에 있다. 여러 시스템에 의해 파
 일을 공유하기 위해서는 반드시 그 파일 시스
 템을 mount하여야 한다. NFS의 가장 큰 단점은
 remote 파일의 semantics를 유지하지 못하여 한
 사용자가 어떤 한 파일을 사용하고 있는 동안
 다른 사용자에게 의하여 그 파일이 지연될 가능
 이 있으며 RFS에 비하여 네트워크 상의 보안메카
 니즘의 기능이 약함에 있다.

SUN과 AT&T는 SystemV에 4.2BSD의 큰 장
 점인 TCP/IP기능 등을 포함시켜 보다 이종 네
 트워크에 다양한 기능을 제공하려 하고 있으며,
 여러 우수한 기능을 보완하여 가까운 장래에 표
 준 이종파일 시스템을 개발하여 사용자들에게
 보다 뛰어난 성능, 유사기종, 다른 기종끼리의
 통신시 보다 우수한 기능을 제공할 것이다.

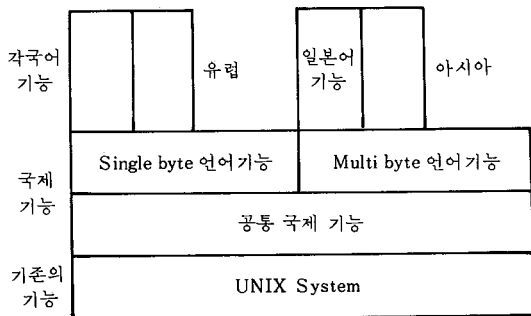
3. UNIX의 국제 표준화

UNIX 시스템이 1960년대 후반에 AT&T Bell

연구소에서 개발되어 보급되어 온 이래 AT&T 이외의 여러 회사, 연구기관들이 독자적인 기능 개발을 시작하면서 UNIX시스템의 장점인 이식성이 결여될 우려가 보였었다. 그리하여 AT&T는 1983년에 System V를 release하고, UNIX시스템의 표준화를 추진하여 왔다.

한편, UNIX 시스템은 미국 뿐만 아니라 유럽, 아시아, 오세아니아의 여러 나라에서도 급속히 전파되어 국제기능이 요구되어져 왔다. AT&T는 이에 부응하여 UNIX 시스템의 표준국제기능과 각국의 고유기능을 나누어 정의하고 UNIX 각국어 기능의 최초 산물로서 JAE Rel. 1.0 (Japanese Application Environment, Rel. 1.0)을 release하게 되었다. 본문에서는 주로 JAE Rel. 1.0을 예로 들어 설명한다.

AT&T는 세계 공통으로 사용되는 표준 UNIX 시스템의 구축에 성공하여, 우선 국제기능을 정의하고 그위에 각국어 기능을 정의하였다. 국제기능이란 각국의 UNIX 시스템 변환시 필요한 공통 기능, 즉 싱글 바이트 언어로서의 공통기능 및 멀티 바이트 언어로서의 공통기능으로 구성된다. 이들의 구성은 (그림 2)와 같이 나타낼 수 있다.



(그림 2) UNIX 시스템의 기능

UNIX 시스템 국제기능에는 4개의 코드 set (1개의 기본 코드와 3개의 확장 코드)가 동시에 사용될 수 있다. 각국의 표준 UNIX 시스템은 그 규칙에 따라서 필요한 언어를 할당해서 사용한다. 따라서 다른 나라의 표준 UNIX 시스템 사이에서도 코드 Set의 모순이 없이 데이터의 전송이 용이하다. UNIX 시스템의 국제기능에

사용되는 코드는 (그림 3)과 같다.

문자Code의 구조

Code set	내부 code
set 0 (ASCII)	0xxxxxxx
set 1	1xxxxxxx 1xxxxxxx or 1xxxxxxx
set 2	SS2 1xxxxxxx 1xxxxxxx or SS2 1xxxxxxx
set 3	SS3 1xxxxxxx 1xxxxxxx or SS3 1xxxxxxx

SS2 : Single Shift2(0x8e) SS3 : Single Shift3(0x8f)

(그림 3) UNIX 시스템 국제기능의 문자코드

UNIX 시스템 일본어 기능에서는 4개의 문자 set를 (그림 4)와 같이 할당하여 사용한다.

UNIX 시스템 일본어 기능에 사용된 code

문 자 집 합	code set	1 byte	2 byte	3 byte
ASCII 문자	0	0xxxxxxx		
JIS 한자	1	1xxxxxxx	1xxxxxxx	
반자 가나	2	SS2	1xxxxxxx	
의자 set (주)	3	SS3	1xxxxxxx	1xxxxxxx
간격 문자	0	00100000		
말소 문자	0	01111111		
기능 문자 0	-	000xxxxx		
기능 문자 1	-	100xxxxx		

(주) JIS C 6226 도형문자 부호표의 빈 영역에 할당되어 사용할 문자는 제외함.

(그림 4) UNIX JAE의 Code 분류

이번에 release되는 UNIX시스템 일본어 기능은 System V JAE Rel. 1.0으로 불리며, UNIX 시스템 상의 사용자 응용 프로그램이 일본어 문자를 포함한 데이터를 처리할 환경에 필요한 기능의 총칭이며, 그 데이터에 제공할 기능은 다음과 같다.

- UNIX 시스템 내에 입력하는 수단
- 사용자 응용 프로그램에의 interface
- 응용 프로그램으로부터 다른 응용 프로그램으로의 pass
- UNIX 시스템으로부터 단말장치 및 다른 컴퓨터 시스템으로의 출력

UNIX System V JAE Rel. 1.0은 UNIX System V Rel. 2.1 AT&T 3B 2 version상에서 가동하는 한 개의 별도 패키지이고, UNIX 시스템의 국제기능과 일본어 기능의 양방을 포함하고 있다. JAE Rel. 1.0의 주요한 기능과 그 각각의 기능을 국제기능과 일본어 기능에 비추어, 다음의 <표 1>로써 나타낸다.

〈표 1〉 UNIX JAE의 기능 및 개요

항 목	개 요	기능 분류																																																																										
Table류	terminfo : 일본어 단말속성의 추가 termio : 일본어 입력 속성의 추가	일본어 기능																																																																										
표준시간	일본 표준시 (JST)의 지정	일본어 기능																																																																										
문자의 입력과 일본어 단말 driver	다음 2개의 방법을 통해서 일본어 문자의 입력을 행할 수 있다. ① 조합 가나 한자 모듈 (module) ② 화면 에디터 (editor) 문자 단위의 가나 한자 변환을 행한다.	일본어 기능																																																																										
확장 문자와 사전	확장 문자와 사전을 작성 보수하는 utility 프로그램 확장 문자 font 작성 프로그램 확장 문자 font load 프로그램 사전 유지보수 프로그램	일본어 기능																																																																										
C언어	JAE Rel. 1. 0에는 literal과 주석에 일본어 문자를 지정할 수 있다.	공통 국제 기능																																																																										
Command	JAE Rel. 1. 0으로 일본어 문자 data를 처리할 command는 다음과 같다. <table border="1" data-bbox="340 714 1053 1195"> <thead> <tr> <th colspan="2">command의 종류</th> <th colspan="4">command 명</th> </tr> </thead> <tbody> <tr> <td colspan="2">shell</td> <td>sh</td> <td colspan="3">shl</td> </tr> <tr> <td colspan="2">editor</td> <td>ed</td> <td colspan="3">jvi</td> </tr> <tr> <td colspan="2">C 언어</td> <td colspan="4">cc</td> </tr> <tr> <td rowspan="4">기타의 Command</td> <td>8bit data 및 8bit 파라미터 전송</td> <td>cat</td> <td>cmp</td> <td>cp</td> <td>diff</td> </tr> <tr> <td></td> <td>echo</td> <td>lp</td> <td>ls</td> <td>mail</td> </tr> <tr> <td></td> <td>mv</td> <td>pg</td> <td colspan="2">pr</td> </tr> <tr> <td></td> <td>wall</td> <td colspan="3">write</td> </tr> <tr> <td rowspan="2">File 명 및 directory 명</td> <td></td> <td>cd</td> <td>chgrp</td> <td>chmod</td> <td>chown</td> </tr> <tr> <td></td> <td>mkdir</td> <td>pwd</td> <td>rm</td> <td>rmdir</td> </tr> <tr> <td>문자 패턴 처리</td> <td></td> <td>awk</td> <td>egrep</td> <td>expr</td> <td>fgrep</td> </tr> <tr> <td></td> <td></td> <td>find</td> <td colspan="3">grep</td> </tr> <tr> <td>베이직 네트 워크</td> <td></td> <td>cu</td> <td>rmail</td> <td>uucp</td> <td>uux</td> </tr> </tbody> </table> <p>JAE Rel. 1. 0으로 support되지 않는 command중에 일본어 문자 데이터를 처리할 수 있는 여러가지의 command가 있다.</p>	command의 종류		command 명				shell		sh	shl			editor		ed	jvi			C 언어		cc				기타의 Command	8bit data 및 8bit 파라미터 전송	cat	cmp	cp	diff		echo	lp	ls	mail		mv	pg	pr			wall	write			File 명 및 directory 명		cd	chgrp	chmod	chown		mkdir	pwd	rm	rmdir	문자 패턴 처리		awk	egrep	expr	fgrep			find	grep			베이직 네트 워크		cu	rmail	uucp	uux	공통 국제 기능 jvi는 일본어
command의 종류		command 명																																																																										
shell		sh	shl																																																																									
editor		ed	jvi																																																																									
C 언어		cc																																																																										
기타의 Command	8bit data 및 8bit 파라미터 전송	cat	cmp	cp	diff																																																																							
		echo	lp	ls	mail																																																																							
		mv	pg	pr																																																																								
		wall	write																																																																									
File 명 및 directory 명		cd	chgrp	chmod	chown																																																																							
		mkdir	pwd	rm	rmdir																																																																							
문자 패턴 처리		awk	egrep	expr	fgrep																																																																							
		find	grep																																																																									
베이직 네트 워크		cu	rmail	uucp	uux																																																																							

또한, UNIX System V JAE 기능의 역할과 앞으로의 방침은 다음과 같다.

아시아의 몇몇 국가들에서는 현재 JAE Rel. 1. 0을 기본으로 하여 멀티 바이트 언어인 자국의 표준 UNIX 시스템을 각각 개발중이다. 이처럼 AT&T로부터 각국어 기능으로서 시작하여 제공된 JAE는 이후, 멀티 바이트 언어의 UNIX 시스템 뿐만아니라 UNIX 시스템 국제기능의 구현을 선도해 갈 역할을 맡고 있다.

JAE Rel.1.0의 release에 의해서 일본어 문자 데이터를 처리함에 있어서 모든 환경이 정비된 의미가 아니고, 필요 최소한의 환경이 제공된다

고 말할 수 있다. AT&T는 금후도 UNIX 시스템 일본어 기능을 계속 확장해 나아갈 것이며 장래 제공될 기능으로는 다음과 같은 것들을 들 수 있다.

- 일본어 메시지 기능
- 일본어 데이터를 처리하는 command수의 확대
- 2바이트 코드의 문자처리 기능
 - C표준 함수
 - Shell의 문자 패턴 처리
 - 정규표현 패턴처리 등