

# UNIX 한글환경

元 裕 憲

弘益大 電子計算學科 教授

## I. 서 론

개방형 구조(open architecture)가 확대됨에 따라 이식성이 뛰어난 운영체제로서 유닉스가 각광을 받고 있다. 앞으로는 유닉스가 워크스테이션 분야는 물론 개인용 컴퓨터 분야까지 표준 운영체제로 자리를 굳힐 것이다.

유닉스 시스템은 1969년 AT&T사 Bell Lab.의 Computer Science Research Center에서 개발한 컴퓨터 운영 체제이다. 초기에는 연구 목적으로 개발되었으며 시간이 지나면서 각 대학과 비영리 교육 기관에 제공되었다. 1980년 초 Berkely 대학에서 유닉스를 VAX에 이식하여 사용하기 시작한 후 여러 공학 분야에서 유닉스를 널리 사용하게 되었다.

그 후 몇몇 회사와 단체에서 유닉스를 서로 다르게 확장해서 규정하고 개발했기 때문에 유닉스 상에서 실행되는 여러 응용 프로그램들이 서로 호환성이 없어지는 문제점이 발생하게 되었다. 그러므로 동일한 기능을 수행하는 응용 프로그램일 지라도 각 제조업체의 유닉스 모델에 따라 재개발해야 하는 중복투자의 불합리를 낳게 되었다. 따라서, 이러한 문제점을 해결하기 위해서 IEEE 산하의 유닉스 표준 단체인 POSIX(Portable Operating System Interface for Computer Environment)와 X/Open 등에서는 표준화를 위한 protability guide를 규정하기 시작하였다.

한편 국내 기업들이 한글 처리 기능을 갖도록 개발하여 공급하고 있는 유닉스 제품들은 한글처리 환경에 대한 표준화 작업이 없었기 때문에 더욱 심각한 문제점을 초래하고 있다. 즉, 동일 영문 유닉스에 한글 환경을 구현한 국내판 한글 유닉스들끼리의 호환

성도 없어지는 등 유닉스 환경의 가장 큰 장점인 호환성 상실이 큰 문제점으로 대두된다.

이와 같은 문제점을 해결하고 국제적 표준화를 위해서 유닉스 운영체제하에서 사용자에게 표준화된 한글처리 환경을 제공할 필요성이 절실히 요구되어 공업진흥청 연구사업으로 유닉스의 한글환경 표준화 작업이 이루어졌다.

이 연구에서 한글 처리 환경 표준은 한글 처리에 필요한 제반 사항들에 관하여 X/Open에서 제안한 자국어 지원 환경하에 통일된 원칙을 적용해서 표준화된 한글처리 기능을 갖는 유닉스 시스템 개발환경 뿐만 아니라 사용자들에게도 표준화된 공통적인 응용 환경을 제공하게 된다. 이에 따라 90년도에 공업진흥청 지원하에 진행된 한글 유닉스 표준화 연구작업으로 작성된 “유닉스 한글환경 표준화”안이 여러 차례의 공청회와 관계 회의를 거쳐 1992년 1월 유닉스 한글 처리 환경에 대한 KS 규격이 확정되었다.

이 표준 규격은 유닉스 운영 체제의 커널 서비스와 명령어 및 라이브러리를 이용하는 사용자들과 한글 유닉스 운영체제를 개발하는 프로그래머들에게 통일된 한글처리 환경에 필요한 제반 사항에 대해서 규정하였다. 또한 추가되는 한글 처리 기능들이 기존의 영문처리 환경에서 사용되던 여러 기능들에 영향을 주지 않도록 하는 것을 원칙으로 하였다. 그리고 이 규격에서 제안한 시스템 인터페이스와 사용자 인터페이스는 모두 문자에 기초한 인터페이스에 국한해서 규정하며, 그래픽 인터페이스에 대해서는 제외하였다. 또한 유닉스 운영체제를 사용하는 컴퓨터 시스템에서 일반적으로 제공되는 유틸리티를 제외한 특수 목적의 응용 프로그램도 이 규격의 범위에서 제외하였다.

이 표준화 연구는 국내 개방형 통신 연구회(OS-IA)와 한국 유닉스 사용자 그룹 산하의 기술 위원회인 SG-OSE의 많은 지원을 받았기에 국내 대부분의 유닉스 지원업체와 사용자 그룹의 의견이 자연스럽게 수렴될 수 있었다.

이 표준 규격을 제정하는데 가장 큰 난항을 겪었던 것은 확장 유닉스 부호인 EUC의 “부호 집합 0”에서 KS C 5636 사용 여부이었다. KS C 5636 자판에서는 “\”문자가 정의된 위치에 “₩”문자를 다시 정의하여 사용하고 있는데, 유닉스에서는 PC와 달리 “\”문자의 사용이 다양하게 보편화 되었으므로 “\” 문자 대신에 “₩”를 사용하는 경우에 국제적인 호환성 문제점과 함께 혼란을 발생시킬 수 있다는 예견이 많이 제기되었다.

여러 번의 관련 회의를 거쳐 기존의 ASCII 문자 집합(ISO 646)을 기본으로 하되 KS C 5636도 사용할 수 있다는 단서를 붙여서, 국내에서는 “\” 문자와 “₩”문자를 선택하여 사용할 수 있도록 규정하였으며, KS C 5601(2바이트 한글 표준 규격)을 EUC의 “부호 집합 1”에서 채택하여 “부호집합 1”을 선택하면 한글 문자를 자연스럽게 사용할 수 있게 하였다. 유닉스 한글 환경 표준 규격에서 한글 처리 환경을 위해 규정된 주요 항목은 다음과 같다.

## II. 일반 규격에 대한 한글처리 기능

### 1. 코드 체계

부호 체계는 유닉스 시스템에서 사용하는 확장 유닉스 부호(extended unix code:EUC)와 문자열 처리에 용이성을 제공하는 2바이트 광역문자(wide character)를 기본체제로 한다. 또한 다중 바이트 문자와 광역문자 사이의 변환에 필요한 함수도 규정하였다.

코드 세트 0에는 ISO 646(정보 교환용 부호, 로마 문자 또는 KS C 5636)으로 하며, 코드세트 1에는 KS C 5601-1987(한국 공업 규격-정보 교환용 부호)로 한다.

확장 유닉스 부호의 표현은 아래 표와 같다.

부호집합	확장 유닉스 부호의 표현	부호 체계
0	0XXXXXXX	〈부표 1〉 또는 KS C 5636
1	1XXXXXXXX1XXXXXXXX	KS C 5601
2	SS2 1XXXXXXX[1XXXXXXX[...]]	미정의
3	SS3 1XXXXXXX[1XXXXXXX[...]]	미정의

부호 집합 2,3의 경우 한글 부호체계가 정의되지 않았으므로 추후 결정시 사용할 수 있다.

### 2. 한국식 표현 규격

새로 개발된 프로그램들이 사용자의 언어 환경과 표현 형식 환경에 대해 독립적으로 설계되기 위해서는 지역화가 필요한 부분들은 수행 시간에 결정이 되도록 하여야 한다. 이를 위해 setlocale() 함수를 사용한다.

setlocale(category, locale)

category:(locale, h)에 정의된 상수로 다음값 중 한값을 갖는다.

- a. LC\_ALL
- b. LC\_CTYPE
- c. LC\_COLLATE
- d. LC\_MONETARY
- e. LC\_NUMERIC
- f. LC\_TIME

#### (1) LC\_COLLATE

사용자가 필요에 따라 정할 수 있도록 하며, 사용자가 특별히 지정하지 않은 경우의 대조순서만을 규정한다. 이는 “C” locale과 동일하다. 즉, 영문자일 경우는 “C Program Locale”에 정해진 순으로 하고, 한글에 대해서는 KS C 5601에 규정된 문자의 배열 순서를 따른다. 그리고 문자 체계의 대조순서는 영문자(single byte), 한글(multi-byte)의 순으로 한다.

정규식에서 두개 이상의 문자 체계로 지정했을 경우 동작은 문자 체계를 기본으로 한다.

#### (2) LC\_CTYPE

한글 유닉스가 갖는 클래스는 국제화되어 통용되는 클래스에 한글 처리를 위해 KS C 5601 문자를 기존의 클래스들에 추가한다.

클래스	내 용
대문자 클래스 (upper class)	C locale의 대문자 클래스 KS C 5601 영문 대문자 클래스 KS C 5601 그리스 대문자 클래스 KS C 5601 라틴 대문자 클래스 KS C 5601 러시아어 대문자 클래스
소문자 클래스 (lower class)	C locale의 소문자 클래스 KS C 5601 영문 소문자 클래스 KS C 5601 그리스 소문자 클래스 KS C 5601 라틴 소문자 클래스 KS C 5601 러시아어 소문자 클래스
문자 클래스 (alpha class)	대문자 클래스+소문자 클래스+ 한글 자모+일본 KANA+한글 +한자

클래스	내 용
십진수 클래스 (digit class)	C locale의 "digit" 클래스
16진수 클래스 (xdigit class)	C locale의 "xdigit" 클래스
스페이스 클래스 (space class)	C locale의 "space" 클래스+ KS C 5601 space
공백 클래스 (blank class)	C locale의 "blank" 클래스
구두 클래스 (punct class)	C locale의 "punct" 클래스
제어 클래스 (cntrl class)	C locale의 "cntrl" 클래스
인쇄 클래스 (print class)	alpha와 space를 포함한 모든 출력 가능한 문자
그래프 클래스 (graph class)	C locale의 그래프 클래스+ KS C 5601의 공백 문자와 필(fill) 문자를 제외한 문자

(3) LC\_MONETARY

원화를 표시하는 화폐 문자는 "₩"를 사용한다.

사용 예) 국제형식 : KRW x,xxx

국내형식 : ₩ ([+]|-) x,xxx

(4) LC\_NUMERIC

(a) 소수점 : "." (마침표-dot, 부호값:00101110)

(b) 천자리 구분자 :

기본 (default) : "." (쉼표-comma, 부호값 :  
00101100)

(a)와 (b)는 문자 체계 0에서 정의된 문자이다.

(5) LC\_TIME

이 규격을 만족하는 시스템에서의 날짜와 시간의 표시는 다음과 같은 순서로 표시되며, 각 항목은 필요에 따라 생략 가능하다.

(a) 표시순서 : Y년 M월 D일 W요일 오전/오후

H시 M분 S초

Y:1<=Y<=9999(년)

M:1<=M<=12(월)

D:1<=D<=31(일)

W:일, 월, 화, 수, 목, 금, 토(요일)

H:0<=H<=23(시간)

M:0<=M<=59(분)

S:0<=S<=61(초)

(참고 1) Y, M(month), D, H, M(minute), S는 digit 클래스에서 정의된 문자이다.

(참고 2) ① 24시간제 (0<=H<=23)와 12시간제

(1<=H<=12) 선택적으로 사용 가능하다.

② S가 61인 경우는 윤초가 삽입된 경우를 나타낸다.

③ Y, M, D, H, M, S : 정수형 변수

④ W:문자형 변수

⑤ 요일 : "W요일"은 약식으로 "(W)"로 표현

⑥ 시간 : "H시 M분 S초"는 "H:M:S"로 표현  
(참고 3) Time Zone: 한국의 표준시를 표시한다.

영문표기 : KST

(6) LANG에 지정할 수 있는 문자열

LANG=ko\_KR.5601

(참 고) 국제화되어 통용되는 예

① language[\_territory[.codeset]]으로 규정

한글 유닉스 : language=ko (ISO 639에 규정됨)  
territory=KR (ISO 3166에 규정됨)

codeset=5601

② 전체길이({NL\_LANGMAX}): 14문자 이하로 규정

3. 메시지 처리 방식

새로 개발하는 프로그램의 생산성 향상과 이식성(protability)을 지원해 주기 위해서는 그림1과 같이 국제화 규격에 알맞게 기존의 언어 종속적인 원시코드를 언어에 비종속적인 원시 코드로 변환시켜 주어야 하고 새로운 환경에 수행할 수 있는 방법을 제공해야 한다.

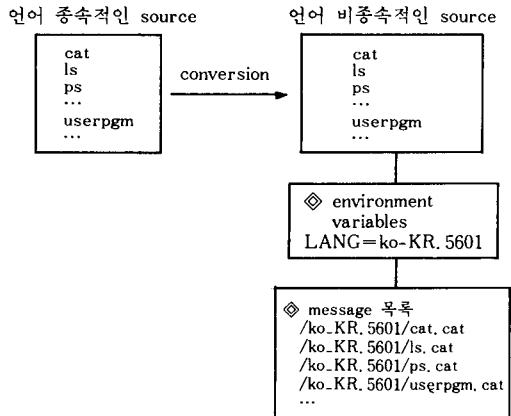


그림 1. 언어 비종속적 수행환경의 예

(메시지 처리)

메시지 처리 방식은 기본적으로 NLSPATH을 이용한 메시지 처리 방식을 이용하며, 이와 관련된 라이브러리도 반드시 제공되어야 한다.

4. 정규식

정규식은 국제화 기능에서 정의된 규칙들을 기본적으로 따른다.

Ⅲ. 상세 규격에 대한 한글처리 기능 규격

1. 명령어 및 유틸리티

명령어 및 유틸리티는 "X/Open Portability Guide Issue 3의 vol.1 XSI Commands and Utilities"를 기본으로 하며, 특히 변경, 삭제, 추가되어야 할 함수들만 소개한다.

여기서 권고대상 함수란 기능이 중복되거나 하드웨어와 소프트웨어를 동시에 제공하는 경우 현재 많이 사용되는 경우를 선택하여 변경을 권장하는 함수이다.

그리고 변경대상 함수는 현재 개발, 사용되는 시스템의 변화로 과거 규정된 규격이 적절하지 않아 변경해야할 함수이다.

① 권고대상 함수

- awk(-f option을 제외)
- cc(-f, -F option을 제외)
- cpio(-p option과 함께 사용되는 "r"을 제외)
- ex(-l option을 제외)
- stty(-tostop/tostop 항목을 제외)

② 변경대상 함수

- du("in 512 byte-units" -> "in number of blocks")
- ed([-s] option을 추가)
- ex([-s]와 [-c command] options를 추가)
- find(-size n[Pc] -> -size n[c], -size에서 "(512 bytes per block)"을 삭제)
- lpstat([id...] 삭제)
- mail("\*" -> "?")
- vi([-c command] option을 추가)

③ 기타 사항

파일명, 디렉토리명, 화일 시스템 이름 및 명령어 이름에 KSC 5601의 모든 문자를 쓸 수 있고, 또한 이들 한글 이름은 8-bit transparent로 동작하는 명

령어 및 유틸리티에서 사용될 수 있다.

2. 시스템 인터페이스 및 헤더 화일

시스템 인터페이스 및 헤더 화일은 "X/Open Portability Guide Issue 3의 vol.2 XSI System Interfaces and Headers"를 기본으로 하며, 특히 한글 기능을 지원하기 위해 변경, 삭제, 추가되어야 할 함수들만 나열한다.

① 삭제 대상 함수

다음의 암호화와 관련된 루틴들은 기초 참조 문서에서는 선택 사항으로서 본 규격에서는 삭제한다.

crypt(), encrypt(), setkey()

② 변경 대상 함수

ⓐ 8-비트 투명성(8-bit transparent)을 갖고 동작하는 함수. (다음 함수는 8-비트 투명성을 제공해야 한다).

advance()	asctime()	closedir()	compile()
ctime()	environ()	fgetc()	fgets()
fopen()	fprintf()	fputc()	fputs()
freopen()	fscanf()	ftw()	getc()
getchar()	getcwd()	getpass()	gets()
loc1	loc2	locs	opendir()
popen()	printf()	putc()	putchar()
putenv()	puts()	readdir()	regexp()
remove()	rewinddir()	scanf()	seekdir()
sprintf()	sscanf()	step()	strcat()
strchr()	strcmp()	strcpy()	strcspn()
strerror()	strlen()	strncmp()	strncmp()
strncpy()	strbrk()	strrchr()	stpspn()
strstr()	strtok()	system()	telldir()
tmpnam()	tmpnam()	ungetc()	vprintf()

ⓑ LC\_NUMERIC과 관련된 함수

(소수점(".")과 천자리 분리자(",")를 처리해야 한다).

atof()	atoi()	atol()	strtod()	strtol()
wcstod()	wcstol()	wcstoul()		

ⓒ LC\_TIME과 관련된 함수

(한국식 날짜, 시간 표현 방법을 처리할 수 있어야 한다).

asctime()	ctime()	strftime()	wcsftime()
-----------	---------	------------	------------

ⓓ 8.4 LC\_COLLATE와 관련된 함수

(사용자가 정의한 문자의 대조순서를 처리할 수 있어야 한다).

strcoll()	strxfrm()
-----------	-----------

④ 추가 대상 함수

한글 입출력과 문자 처리를 위한 기능을 통일성 있게 제공하기 위해서는 광역 문자에 관련된 함수들

을 제공하여야 한다. 관련 함수의 이름들은 XPG4의 draft2와 ISO DIS 9899(ANSI X3.159)를 기초로 참조 문서로 하고 여기에 한글 처리 함수를 추가한다. 이렇게 함으로써 C 언어 수준에서의 응용 프로그램 호환성을 제공한다.

㉓ 새로 지원해야 할 한글 입출력 및 문자 처리함수

catelose()	catgets()	catopen()	fgetwc()
fgetws()	fputwc()	fputws()	getwc()
getwchar()	getws()	iconv()	iconv_close()
iconv_open()	is_wctype()	iswalnum()	iswalphabeta()
iswcntrl()	iswctype()	iswdigit()	iswgraph()
iswlower()	iswprint()	iswpunct()	iswspace()
iswupper()	iswxdigit()	localeconv()	mblen()
mbstowcs()	mbtowc()	nl_langinfo()	perror()
putwc()	putwchar()	putws()	set_wctype()
setlocale()	strerror()	strfmon()	strptime()
towlower()	towupper()	ungetwc()	wcsat()
wcschr()	wsscmp()	wscoll()	wscpy()
wcslen()	wcsncat()	wcsncmp()	wcsncpy()
wcsncpy()	wcsprbrk()	wcsrchr()	wcsspn()
wcstok()	wcstombs()	wcswcs()	wcswidth()
wcsxfrm()	wctomb()	wcwidth()	

3. 라이브러리

라이브러리는 유닉스 시스템에서 제공하는 curses 라이브러리만을 대상으로 하며, X/Open Portability Guide Issue 3의 volume III XSI Supplementary Definitions (ISBN:0-13-685850-3)을 기본으로 따르며, 변경되는 함수만 제시한다. 또한 관계없는 라이브러리 함수는 ISBN:0-13-685850-3을 따르며, 한글 사용의 윈도우 입출력 명령은 추가로 별도로 작성하고 이로 인하여 일부 함수는 변경된다.

① 한글 입출력 지원을 위하여 추가되는 함수

Curses 기본 함수에 한글을 지원하기 위하여 광역문자라는 개념으로 함수명 중간에 w를 삽입하여 새로운 라이브러리 함수명을 정하며, 주요함수는 다음과 같다.

addwch()	waddwch()	mvaddwch()	mvwaddwch()
inswch()	winswch()	mvinswch()	mvwinswch()
getwch()	wgetwch()	mvgetwch()	mvwgetwch()
addwstr()	waddwstr()	mvaddwstr()	mvwaddwstr()
getwstr()	wgetwstr()	mvgetwstr()	mvwgetwstr()

② 변경되는 함수

• 한글 자료를 받을 수 있도록 %wc, %ws를 지원해야 하는 함수

printw()	wprintw()	mvprintw()	mvwprintw()
scanw()	wscanw()	mvscanw()	mvwscanw()

• 한글 경우에도 문자 단위의 삭제 기능을 지원해야 하는 함수

delch()	wdelch()	mvdelch()	mvwdelch()
---------	----------	-----------	------------

• Box 함수에서 경계 문자로 한글을 처리하도록 광역문자를 쓸 수 있다.

IV. 결론 및 기대효과

한글 유닉스 표준화 작업에서 수행된 연구는 POSIX와 X/OPEN에서 제안한 국제규격 지침안에 가능한 저촉되지 않으면서 한글 환경을 자연스럽게 구축할 수 있도록 하였다. 이 연구는 유닉스 운영체제의 커널 서비스와 명령어 및 라이브러리를 이용하는 사용자들과 한글 유닉스 운영체제를 구축하는 개발자들에게 통일된 한글 처리 환경을 제공하는데 필요한 환경에 대해 규정하였다. 또한 이 연구는 이론 중심이 아니라 기존의 한글 유닉스 개발팀들의 참여 하에 이루어져 실제 응용이 될 수 있도록 작성되어 다음과 같은 기대 효과를 갖는다.

- ◎ 기존의 한글 유닉스의 사용 환경을 통일시킴으로써 한글 유닉스 사용자에게 편의를 제공함과 동시에 생산성을 향상시킴
- ◎ 유닉스 한글 환경이 표준화됨으로써 한글 유닉스에서 개발된 소프트웨어 호환성 증가
- ◎ 유닉스의 표준화된 한글 환경을 개발 업체들에게 제공함으로써 한글 유닉스 개발 생산성 향상을 꾀함
- ◎ 외국인에 의한 표준 한글 유닉스 사양을 배제함
- ◎ 한글 유닉스 국내 사용을 저변화시킴

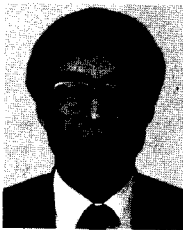
參 考 文 獻

[ 1 ] 유닉스 한글 환경 보고서, 1992.  
 [ 2 ] Alan Deikman, "UNIX Programming on the 80286/80386", M&T Publishing, Inc., 1988.  
 [ 3 ] AT&T, "UNIX SYSTEM V/386 Release 3.2 Streams Primer", Prentice-Hall, Inc., 1989.  
 [ 4 ] Keith Havilland & Ben Salama, "UNIX System Programming", Addison-Wesley Publishing Company, 1987.  
 [ 5 ] Maurice J. Bach, "The Design of The UNIX Operating System", Prentice Hall, Inc., 1986.

- [ 6 ] Mitchell Waite, "UNIX Papers for UNIX Developers and Power Users", Howard W. Sams & Company, 1987.
- [ 7 ] R.S Tare, "UNIX Utilities, a Programmer's Reference", McGraw-Hill Book Company, 1987.
- [ 8 ] Uniform, "Your Guide to POSIX, Updated", 1989.
- [ 9 ] Uniform, "POSIX Updated : Shell and Utilities", 1989.
- [10] "XSI Commands, and Utilities Classification According to Internationalization and Localization Issues"
- [11] X/OPEN Company, "X/OPEN Portability Guide",  
vol. 1 XSI Commands, and Utilities,  
vol. 2 XSI System Interface and Headers,  
vol. 3 XSI Supplementary Definitions, Prentice Hall, Inc., 1988.
- [12] "UNIX SYSTEM V Multi-National Language Supplement Release 3.2 Product Overview", AT&T UNIX Software Operation Pacific, 1989.
- [13] " '90 KUUG UNIX Symposium Proceedings", 한국 유닉스 사용자그룹, 1990.
- [14] "표준 한글 한자 코드 확장 및 처리방식 연구 진행보고서", 한국 표준연구소, 1990. 5~1990.8



## 筆者紹介



元 裕 憲

1948年 12月·10日生

1972年 2月 성균관대 수학과 졸업(학사)

1975年 8月 한국과학기술원 전자계산학과(석사)

1985年 8月 고려대(이학박사)

1975年~1976年 한국과학기술연구소 연구원

1986年~1987年 R. P. I 객원 교수

1976年~현재 홍익대학교 전자계산학과 교수

주관심분야 : 컴파일러, 프로그래밍 언어디자인, 소프트웨어 공학, 분산언어,  
객체 지향언어, 하드웨어 기술언어 등.