



# 국제 언어 환경 설명서

---

Sun Microsystems, Inc.  
4150 Network Circle  
Santa Clara, CA 95054  
U.S.A.

부품 번호: 816-3980-10  
2002년 5월

Copyright 2002 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. 모든 권리는 저작권자의 소유입니다.

본 제품과 문서는 저작권에 의해 보호되며 그 사용, 복사, 배포 및 발체를 제한하는 라이선스에 의거하여 배포됩니다. Sun과 Sun이 사용을 허가한 자의 사전 서면 허가 없이는 본 제품이나 문서의 일부나 전체를 어떠한 형식으로도 복제할 수 없습니다. 글꼴 기술을 포함한 협력업체 소프트웨어는 Sun의 공급업체에서 저작권을 소유하고 있으며, 사용 허가를 받았습니다.

본 제품의 일부는 캘리포니아 대학교에서 사용 허가를 받은 Berkeley BSD 시스템으로부터 파생되었을 수 있습니다. UNIX는 미국 및 다른 국가에서 등록된 상표로서, X/Open Company, Ltd.를 통해서 독점적으로 사용 허가를 받았습니다.

Sun, Sun Microsystems, Sun 로고, docs.sun.com, AnswerBook, AnswerBook2, Java, XView, ToolTalk, Solstice AdminTools, SunVideo SunOS, Solaris, X11, SPARC, UNIX, PostScript, OpenWindows, AnswerBook, SunExpress, SPARCprinter, JumpStart, Xlib 및 Solaris는 미국 및 다른 국가에서 Sun Microsystems, Inc.의 상표, 등록상표 또는 서비스 상표입니다. 모든 SPARC 상표는 사용 허가를 받았으며 미국 및 다른 국가에서 SPARC International, Inc.의 상표 또는 등록상표입니다. SPARC 상표를 사용하는 제품은 Sun Microsystems, Inc.가 개발한 구조를 기반으로 하고 있습니다.

OPEN LOOK 및 Sun™ 그래픽 사용자 인터페이스는 Sun Microsystems, Inc.가 사용자와 허가받은 업체를 위해 개발했습니다. Sun은 컴퓨터 산업을 위한 시각적 또는 그래픽 사용자 인터페이스 개념의 연구 개발에 있어서 Xerox가 이룩한 선구적인 노력을 인정합니다. Sun은 Xerox로부터 Xerox Graphical User Interface에 대한 비독점적 사용 허가를 받았으며, 이 사용 허가는 OPEN LOOK 그래픽 사용자 인터페이스를 실행하며 Sun의 서면 사용 허가 계약을 준수하는, Sun으로부터 사용 허가를 받은 사용자들에게도 적용됩니다.

연방 정부 취득: 상업용 소프트웨어-미국 정부 사용자는 기본 라이선스 내용 및 조건을 준수해야 합니다.

설명서는 "있는 그대로" 제공되며, 상품성, 특정 용도에 대한 적합성 또는 비침해에 대한 묵시적인 보증을 비롯한 일체의 명시적 또는 묵시적 조건, 표현 및 보증 책임이 없음을 선언합니다. 단, 이러한 면책문이 법적으로 효력이 없는 경우에는 예외입니다.

---

Copyright 2002 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Tous droits réservés.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées du système Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, docs.sun.com, AnswerBook, AnswerBook2, et Solaris sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" ET AUCUNE GARANTIE, EXPRESSE OU IMPLICITE, N'EST ACCORDEE, Y COMPRIS DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DE LA PUBLICATION A REpondre A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ELLE NE SOIT PAS CONTREFAISANTE DE PRODUIT DE TIERS. CE DENI DE GARANTIE NE S'APPLIQUERAIT PAS, DANS LA MESURE OU IL SERAIT TENU JURIDIQUEMENT NUL ET NON AVENU.



030309@5533



# 목차

---

머리말 15

- 1 Solaris 국제화 개요 21
  - 새로운 국제화 및 지역화 기능 22
  - 국제화 및 지역화의 개념 23
    - 국제화 기본 절차 23
    - Solaris 인터페이스의 지역화 기능 24
  - 로캘이란? 25
    - 전체 및 부분 로캘 26
    - 로캘의 영향을 받는 동작 26
    - 로캘 범주 27
  - 지역화용 로캘 범주 사용 27
    - 시간 형식 28
    - 날짜 형식 28
    - 숫자 29
    - 통화 30
  - 언어 단어 및 글자 차이 32
    - 단어 분리자 32
    - 정렬 순서 32
    - 문자 세트 32
  - 키보드 차이점 35
  - 용지 크기의 차이 35
- 2 일반 국제화 기능 37
  - 코드 세트 독립 지원 37

CSI 접근 방법	38
CSI화 된 명령	38
Solaris 9 CSI화 된 라이브러리	39
로켈 데이터베이스	39
프로세스 코드 형식	40
멀티바이트 지원 환경	40
동적으로 링크된 응용프로그램	41
변경된 인터페이스	41
ctype 매크로	42
libc의 국제화 API	43
genmsg 유틸리티	50
사용자 정의 및 사용자 확장 가능 코드 변환	51
<b>3 Solaris 9 환경에서 지역화</b>	<b>53</b>
지역화를 위한 소프트웨어 지원	53
Solaris 9 로켈 패키지 요약	53
본 릴리스에서 추가 로켈	54
지원되는 로켈	55
로켈용 복수 키 구성 시퀀스	61
Solaris 9 제품에서의 키보드 지원	62
SPARC 시스템에서 키보드 변경하기	63
Intel 시스템에서 키보드 변경하기	66
<b>4 지원되는 아시아권 로켈</b>	<b>75</b>
아시아권에서 지원되는 로켈	75
간체 및 정체 한자용 입력 방식 보조 창	76
태국어 지역화	77
태국어 입력 방식 보조 창	78
간체 한자 지역화	78
정체 한자 지역화	81
일본어 지역화	84
일본어 로켈	84
일본어 문자 세트	84
일본어 글꼴	84
일본어 입력 시스템	85
일본어 단말기용 단말기 설정	86
일본어 iconv 모듈	86

사용자 정의 문자 지원	86
부분 로캘과 전체 로캘의 차이	87
한국어 지역화	87
<b>5 UTF-8 로캘 지원의 개요</b>	<b>91</b>
Unicode의 개요	91
Unicode 로캘: en_US.UTF-8 지원 개요	92
데스크탑 입력 방식	94
스크립트 선택 및 입력 모드	95
입력 방식 액세스하기	95
정체 한자 입력 모드	111
정체 한자(홍콩) 입력 모드	111
Unicode 16진법 입력 모드	111
테이블 조회 입력 모드	112
시스템 환경	112
로캘 환경 변수	112
TTY 환경 설정	113
코드 변환	116
DtMail	117
프로그래밍 환경	119
X 응용프로그램과 함께 사용되는 FontSet	120
CDE/Motif 응용프로그램의 글꼴 목록 정의	120
<b>6 Complex Text Layout</b>	<b>121</b>
CTL 기술의 개요	121
CTL 구조의 개요	122
X 라이브러리 기반 응용프로그램에 대한 CTL 지원	122
New XOC 자원	122
CTL 기술 지원을 위한 Motif 변경 사항	123
XmNlayoutDirection	124
레이아웃 방향 결정하기	124
XmStringDirection	125
XmRendition	125
추가 레이아웃 동작	126
XmText 및 XmTextField	126
문자 방향 활동 루틴	127
문자 방향 추가 동작	127

XmText 활동 루틴	128
XmTextFieldGetLayoutModifier	130
XmTextGetLayoutModifier	131
XmTextFieldSetLayoutModifier	131
XmTextSetLayoutModifier	132
XmStringDirectionCreate	132
UIL Arguments	133
CTL 응용프로그램 개발 방법	133
레이아웃 방향	133
Rendition 만들기	135
Rendition 편집하기	135
리소스 파일에서 렌더 테이블 만들기	136
응용프로그램에서 렌더 테이블 만들기	137
수평 탭	138
마우스 선택	139
키보드 선택	140
텍스트 자원 및 기하학	140
이식 관련 지침	141
<b>7 mp를 사용한 인쇄 필터 강화</b>	<b>143</b>
UTF-8에 대한 인쇄	143
mp 인쇄 필터 강화 개요	144
로컬 특정 구성 파일 mp.conf와 함께 mp 사용하기	144
로컬 특정 포스트스크립트 프롤로그 파일과 함께 mp 사용하기	145
Xprt(X 인쇄 서버) 클라이언트로 mp 사용하기	145
구성 파일의 지역화	145
기존 프롤로그 파일 사용자 정의 및 새 프롤로그 파일 추가하기	151
포스트스크립트 파일 사용자 정의	151
.xpr File Customization	154
새 .xpr 파일 작성하기	157

A	iconv 코드 변환	159
B	Software CD의 부분 로컬 패키지 목록	187
C	Language CD의 전체 로컬 패키지 목록	193
	색인	213





## 표

---

표 1-1	국제 시간 형식	28
표 1-2	국제 날짜 형식	28
표 1-3	국제 숫자 규약	29
표 1-4	국제 통화 규약	30
표 1-5	유로 통화를 지원하는 사용자 로케일	31
표 1-6	독일어 로케일과 관련 LC_MONETARY	31
표 1-7	공통 국제 페이지 크기	35
표 2-1	libc의 메시지 처리 함수	43
표 2-2	libc의 코드 변환	44
표 2-3	libc의 정규 표현식	44
표 2-4	libc의 넓은 문자 클래스	44
표 2-5	libc의 로케일 수정 및 질의	44
표 2-6	libc의 로케일 데이터 질의	45
표 2-7	libc의 문자 분류 및 자역	45
표 2-8	libc의 문자 조합	46
표 2-9	libc의 통화 형식	46
표 2-10	libc의 날짜 및 시간 형식	47
표 2-11	libc의 멀티바이트 처리	47
표 2-12	libc의 넓은 문자 및 문자열 처리	47
표 2-13	libc의 형식화된 넓은 문자 입출력	48
표 2-14	넓은 문자열libc	49
표 2-15	libc의 넓은 문자 입출력	49
표 3-1	아시아	55
표 3-2	남양주	56
표 3-3	중앙 아메리카	56
표 3-4	중부 유럽	57

표 3-5	동유럽	57
표 3-6	중동	58
표 3-7	북아프리카	58
표 3-8	북아메리카	58
표 3-9	북유럽	59
표 3-10	남아메리카	59
표 3-11	남부 유럽	60
표 3-12	서유럽	60
표 3-13	Compose 키로 만든 구분 기호	61
표 3-14	지역 키보드 지원	62
표 3-15	유형 4,5 및 5c 키보드용 레이아웃	64
표 4-1	아시아권 로캘 요약	75
표 4-2	zh_CN.EUC 로캘용 트루 타입 글꼴	79
표 4-3	zh_CN.EUC 로캘용 비트맵 글꼴	79
표 4-4	zh_CN.GBK 로캘용 트루 타입 글꼴	79
표 4-5	zh_CN.GBK 로캘용 비트맵 글꼴	80
표 4-6	zh_CN.GB18030 로캘용 트루 타입 글꼴	80
표 4-7	zh_CN.GB18030 로캘용 비트맵 글꼴	80
표 4-8	간체 한자용 코드 세트 변환	80
표 4-9	zh_TW 로캘용 정체 한자 트루 타입 글꼴	82
표 4-10	zh_TW 로캘용 정체 한자 비트맵 글꼴	82
표 4-11	zh_HK.BIG5HK 로캘용 트루 타입 글꼴	82
표 4-12	zh_HK.BIG5HK 로캘용 비트맵 글꼴	83
표 4-13	정체 한자용 코드 세트 변환	83
표 4-14	일본어 비트맵 글꼴	85
표 4-15	일본어 트루 타입 글꼴	85
표 4-16	ko 로캘용 Solaris 9 한국어 비트맵 글꼴	88
표 4-17	ko.UTF-8 로캘용 Solaris 9 한국어 비트맵 글꼴	88
표 4-18	ko/ko.UTF-8 로캘용 Solaris 9 한국어 트루 타입 글꼴	89
표 4-19	한국어 iconv	89
표 5-1	입력 모드 스위치 키 시퀀스	97
표 5-2	공통 라틴-1 Compose 키 시퀀스	98
표 5-3	공통 라틴-2 Compose 키 시퀀스	102
표 5-4	공통 라틴-3 Compose 키 시퀀스	104
표 5-5	Common 라틴-5 Compose 키 시퀀스	105
표 5-6	공통 라틴-9 Compose 키 시퀀스	105
표 5-7	en_US.UTF-8이 지원하는 32비트 STREAMS 모듈	113
표 5-8	en_US.UTF-8이 지원하는 64비트 STREAMS 모듈	113

표 6-1	XmRendition의 새로운 자원	125
표 6-2	Xm CTL의 새로운 자원	127
표 6-3	UIL	133
표 7-1	선택적 키워드/값 쌍	149
표 A-1	사용 가능한 유니코드 관련 iconv 코드 변환 모듈	159
표 A-2	사용 가능한 유니코드 및 IBM/Microsoft EBCDIC 및 PC 코드 페이지 관련 iconv 코드 변환 모듈	180
표 A-3	UTF-8에 대한 사용 가능한 iconv 코드 변환 - IBM 및 Microsoft EBCDIC/PC 코드 페이지	182
표 B-1	부분 로캘의 목록	187
표 C-1	간체 한자	193
표 C-2	프랑스어	195
표 C-3	독일어	197
표 C-4	이탈리아어	198
표 C-5	일본어	200
표 C-6	한국어	203
표 C-7	스페인어	205
표 C-8	스웨덴어	207
표 C-9	정체 한자	208
표 C-10	아시아권 공유	211



## 그림

---

그림 1-1	Solaris 운영 환경의 로컬 기능과 구조	23
그림 3-1	아랍어 키보드	66
그림 3-2	벨기에 키보드	67
그림 3-3	키릴어(러시아어) 키보드	67
그림 3-4	덴마크어 키보드	67
그림 3-5	핀란드어 키보드	68
그림 3-6	프랑스어 키보드	68
그림 3-7	독일어 키보드	68
그림 3-8	이탈리아어 키보드	69
그림 3-9	일본어 키보드	69
그림 3-10	한국어 키보드	69
그림 3-11	네덜란드(네덜란드어) 키보드	70
그림 3-12	노르웨이어 키보드	70
그림 3-13	포르투갈어 키보드	70
그림 3-14	스페인어 키보드	71
그림 3-15	스웨덴어 키보드	71
그림 3-16	스위스(프랑스어) 키보드	71
그림 3-17	스위스(독일어) 키보드	72
그림 3-18	정체 한자 키보드	72
그림 3-19	터키어 F 키보드	72
그림 3-20	터키어 Q 키보드	73
그림 3-21	영국 키보드	73
그림 3-22	미국 키보드	73
그림 3-23	미국/UNIX 키보드	74
그림 4-1	보조 창 지원용 인터페이스 모델	77
그림 5-1	입력 모드 선택 창	95

그림 5-2	아랍어 키보드	105
그림 5-3	키릴 문자(러시아어) 키보드	106
그림 5-4	그리스어 유럽 키보드	107
그림 5-5	그리스어 UNIX 키보드	107
그림 5-6	헤브루어 키보드	108
그림 5-7	힌디어 키보드	108
그림 5-8	힌디어-Shift 키보드	108
그림 5-9	일본어 키보드	109
그림 5-10	한국어 키보드	110
그림 5-11	태국어 키보드	110
그림 5-12	DtMail 새 메시지 창	118
그림 6-1	CTL 구조	122
그림 6-2	레이아웃 방향	134
그림 6-3	탭 이동 동작	138

## 머리말

---

국제 언어 환경 설명서에서는 Solaris™ 9 운영 환경의 새로운 국제화 기능에 대해 설명합니다. 다양한 언어와 문화적 규약을 지원하는 국제적인 소프트웨어 제품을 제작하기 위한 본 릴리스의 사용법에 관한 중요한 정보가 포함되어 있습니다.

본 책에서는 국제적 응용프로그램 개발 및 전 세계 언어 서비스의 관리를 용이하게 하기 위해 Solaris 플랫폼에서 제공하는 구체적인 기능과 함께 언어 활성화와 관련된 기본 속성을 설명합니다.

특히, 머리말에는 다음 정보가 포함되어 있습니다:

- “본 설명서의 사용 대상” 15 페이지
- “설명서 구성” 16 페이지
- “관련 서적과 사이트” 16 페이지

필요할 경우 이 설명서는 본 릴리스의 국제화 기능의 추가 또는 보다 자세한 정보를 포함한 설명서 세트의 다른 설명서도 안내해 줍니다. Sun 설명서의 온라인 주문법과 본 설명서에서 사용된 활자체 규약의 포인터를 얻게 됩니다.

---

## 본 설명서의 사용 대상

본 설명서는 Solaris 9 운영 환경을 위한 국제적 제품 및 응용프로그램을 설계하는 소프트웨어 개발자 및 관리자를 위해 작성되었습니다.

본 설명서에서는 C 프로그래밍 언어에 대한 지식을 갖고 있는 것으로 가정합니다.

모든 운영 체제 정보는 SunOS™ 5.9 운영 환경에 준합니다.

---

## 설명서 구성

본 설명서의 각 장은 다음과 같이 구성되어 있습니다:

- 제 1 장에서는 Solaris 9의 새로운 국제화 및 지역화 기능을 설명하며 몇 개 국가에서 사용하는 유로(€)에 대해 소개합니다.
- 제 2 장에서는 Solaris 9 제품용 libc의 코드 세트 독립성, CSI 및 API 지원에 대해 설명합니다.
- 제 3 장에서는 다국어 Solaris 제품의 지역화, 19종의 새 키보드를 포함한 새로운 키보드 지원을 포함한 Solaris 9의 콘텐츠에 대해 설명합니다.
- 제 4 장에서는 아시아권에서 지원되는 로캘, 입력 시스템 및 문자 지원에 대해 설명합니다.
- 제 5 장에서는 본 릴리스에 내장된 en\_US.UTF-8 로캘과 국제화 기능에 대해 설명합니다. 일본어, 한국어, 간체 및 정체 한자 입력 모드와 함께 키릴 자모, 그리스어, 히브리어, 힌디어, 태국어 입력 방식이 포함됩니다.
- 제 6 장에서는 Motif API가 아랍어, 히브리어 및 태국어와 같은 논리적 및 물리적 텍스트 표현 간의 복잡한 변환이 필요한 쓰기 시스템을 지원하도록 해 주는 CTL (Complex Text Layout) 확장에 대해 설명합니다.
- 제 7 장에서는 유럽어 및 아시아권 언어의 인쇄와 mp(1) 인쇄 필터의 개선 사항에 대한 구체적인 정보와 함께 Solaris 9 운영 환경에서의 인쇄 지원에 대해 설명합니다.
- 부록 A에는 사용 가능한 iconv 변환의 표 목록이 있습니다.
- 부록 B에는 OS CD의 부분 지역화 패키지 이름 표가 있습니다.
- 부록 C에는 Language CD의 언어 패키지를 나타내는 표가 있습니다. 간체 한자, 프랑스어, 독일어, 이탈리아어, 일본어, 한국어, 스페인어, 스웨덴어, 정체 한자 및 공유용 표가 있습니다.

---

## 관련 서적과 사이트

### Sun 글로벌 응용프로그램 개발자 코너

개발자들의 응용프로그램 세계화를 위한 정보는 Sun 글로벌 응용프로그램 개발자 코너(Sun GADC)를 참조하십시오.

Sun 글로벌 응용프로그램 개발자 코너는 이전에 릴리스된 Sun 글로벌 응용프로그램 개발자 키트 1.0의 갱신된 웹 버전입니다. 사이트의 주소는 다음과 같습니다:

<http://www.sun.com/developers/gadc>



Sun 글로벌 응용프로그램 개발자 코너에는 국제적 호환성 및 문제 해결용 테스트를 포함한 글로벌 소프트웨어 개발 과정에서 생기는 다양한 설계 및 개발 관련 문제들을 다루는 종합적인 국제화 도구와 설명서가 포함되어 있습니다.

이 사이트에는 개발자가 의사 영어를 사용해 자신의 국제화된 응용프로그램을 테스트할 수 있게 하는 Sun 멀티바이트 영어(MBE) 로케일과 같은 테스트용 도구들이 포함되어 있습니다. 이 기능은 특정 모국어로 개발된 자신의 응용프로그램을 테스트하고자 하는 영어를 사용하는 개발자들에게 유용합니다. Sun 멀티바이트 영어 로케일은 무료로 다운로드해 사용할 수 있습니다. 그 밖에 유용한 자료로 참조용 예제 및 C 코드, Solaris 운영 환경의 국제 언어 지원에 대한 백서, 기술 관련 기사 및 빠른 참조를 위해 유용한 세계화 링크 등이 포함됩니다. 제품이 국제화 평가를 위해 개발자가 사용할 수 있는 확인 목록과 Sun 국제화 관련 질의를 질문할 수 있는 연락처 페이지가 있습니다.

## Java 개발 키트

Java 개발 키트에 관한 정보는 다음 사이트를 참조하십시오.

<http://java.sun.com/j2se/1.3/docs/guide/intl/index.html>

## Solaris 공통 데스크탑 환경

*Solaris Common Desktop Environment: Programmer's Guide* 또한 Solaris Documentation CD에 포함되어 있는 CDE 개발자 모음의 일부입니다.

## OSF/Motif 정보

*OSF/Motif Programmer's Guide, Release 1.2* Englewood Cliffs, New Jersey, Prentice-Hall, 1993. Open Software Foundation's (OSF) Guide에서는 OSF/Motif 응용프로그램 프로그래밍 인터페이스를 사용해 Motif 응용프로그램을 만드는 방법에 대해 설명합니다. 이 설명서에서는 Motif 위젯 세트 아키텍처 개요를 제시하며 Motif 툴킷에 대해 설명하고 Motif 응용프로그램의 모델과 예제를 제공합니다.

## 포스트스크립트 정보

이 설명서 모음은 포스트스크립트 응용프로그램을 성공적으로 개발하는데 필수적입니다.

*PostScript Language Reference Manual (Second Edition)*은 포스트스크립트용 표준 참조용 작업입니다. 모든 연산자, Display PostScript (DPS), 레벨 1과 레벨 2의 개념을 정의하는 설명서입니다. 이 책은 장치 독립적 인쇄 언어로서 포스트스크립트의 기초적인 부분을 다룹니다. 포스트스크립트의 글꼴 및 문자 처리를 위한 특수 기능이 설명되어 있습니다. 이 책의 부록 E에서는 표준 문자 세트와 코드화 벡터도 설명합니다. 인터프리터에 내장되어 있거나 다른 업체에서 제공한 글꼴의 구성에 대해서도 논의합니다.

*Programming the Display PostScript System with X*는 X 윈도우 및 Display PostScript를 사용하는 응용프로그램 개발자를 위한 것입니다. 이 책에서는 Display PostScript를 사용해 화면 표시 및 프린터 출력용 정보를 만드는 응용프로그램들을 작성하는 방법을 설명합니다. 코딩 기법에 대해 자세히 설명합니다.

X 윈도우 시스템이 X Display PostScript 시스템(X/DPS)으로 확장되었습니다. 이 시스템은 클라이언트측에서는 응용프로그램 호출 가능 라이브러리를, X 서버측에서는 관련 확장자를 사용합니다.

## Solaris 운영 환경에서의 국제화 프로세스

Tuthill, Bill 및 David Smallberg. *Creating Worldwide Software: Solaris International Developer's Guide*, 2판. Mountain View, California, Sun Microsystems Press, 1997. books.sun.com과 www.sun.com/books/를 참조하십시오. 이 책은 Solaris 운영 환경에서의 국제화 프로세스에 대한 일반적 개요를 제공합니다.

---

## Sun 문서 온라인 액세스

docs.sun.com<sup>SM</sup> 웹사이트를 통해 Sun 온라인 기술 설명서에 액세스할 수 있습니다. docs.sun.com 아카이브를 찾아보거나 특정 책 제목 또는 주제를 검색할 수 있습니다. URL은 <http://docs.sun.com>입니다.

---

## 활자체 규약

다음 표는 이 책에서 사용된 활자체에 대해 설명합니다.

표 P-1 활자체 규약

서체 또는 기호	의미	예
AaBbCc123	명령어, 파일 및 디렉토리의 이름; 화면상의 컴퓨터 출력	사용자의 .login 파일을 편집하십시오.  모든 파일을 나열하려면 ls -a를 사용하십시오.  machine_name% you have mail.

표 P-1 활자체 규약 (계속)

서체 또는 기호	의미	예
<b>AaBbCc123</b>	컴퓨터 화면상 출력과 대비되는 사용자 입력	machine_name% <b>su</b> 암호:
<i>AaBbCc123</i>	명령줄에서의 대체 기호. 실제 이름이나 값으로 대체됩니다.	파일을 삭제하려면 <b>rm</b> <i>filename</i> 을 입력하십시오.
<i>AaBbCc123</i>	책 제목, 새로운 단어나 용어 또는 강조할 단어	<b>사용 설명서</b> 6장을 읽으십시오. 이를 <b>클래스</b> 옵션이라고 합니다. 이 작업을 수행하려면 <b>루트</b> 여야 합니다.

## 명령 예제에서의 셸 프롬프트

다음 표는 C 셸, 본 셸 및 콘 셸의 기본 시스템 프롬프트와 슈퍼유저 프롬프트를 보여줍니다.

표 P-2 셸 프롬프트

셸	프롬프트
C 셸 프롬프트	machine_name%
C 셸 슈퍼유저 프롬프트	machine_name#
본 셸 및 콘 셸 프롬프트	\$
본 셸 및 콘 셸 슈퍼유저 프롬프트	#



## Solaris 국제화 개요

---

이 절에서는 국제화 및 지역화에 관한 일부 일반 정보에 대해 논의합니다.

- “새로운 국제화 및 지역화 기능” 22 페이지
- “국제화 및 지역화의 개념” 23 페이지
- “로케이란?” 25 페이지
- “지역화용 로케일 범주 사용” 27 페이지
- “언어 단어 및 글자 차이” 32 페이지

Solaris 9 제품에는 선택된 로케일로 유니코드 및 ISO/IEC 10646에 규정되어 있는 전체 유니코드 3.1 지원이 포함되어 있습니다. Solaris 9 릴리스는 Sun의 국제 시장을 위한 주요 릴리스입니다. 여기에는 몇 가지 새로운 기능이 포함되어 있습니다.

Solaris 9 운영 환경은 처음부터 세계 각국의 언어를 이용할 수 있도록 설계되었습니다. 국제화에 대해 플러그 가능, 서비스 기반 방식으로 접근함으로써 Solaris 국제화 아키텍처는 전 세계적인 응용프로그램 및 언어 서비스의 개발, 배치 및 관리를 쉽게 해 줍니다. 간편한 단일 다국어 제품을 통해 사용자들은 태국어 및 힌디어를 지원하기 위해 필요한 복잡한 텍스트 레이아웃 환경과 아랍어 및 히브리어와 같은 언어를 위한 양방향 레이아웃 환경을 포함해 39개 언어와 162개 로케일에 대한 폭 넓은 지원의 혜택을 누리게 됩니다.

Solaris 국제화 아키텍처는 입력 방식, 문자 세트 코드화, 코드 세트 변환 및 언어 서비스의 기본적인 측면들을 처리하는 유연하고 플러그 가능한 메소드를 제공합니다. 이미 제공된 강력한 도구들을 사용하거나 자신의 환경을 사용자 정의할 수 있습니다. 입력 방식의 작동 방식이나 활성화해야 할 코드 세트 변환기에 대한 지식이 없어도 간단히 표준 API로 여러 개의 언어 환경에서 응용프로그램을 배치할 수 있습니다. 또한 특정 언어 속성을 사용자 정의할 수도 있습니다. 이 아키텍처는 사용자의 변환기 테이블 변경 또는 새 입력 방식 편집기 추가가 가능합니다.

Solaris X 국제화 프레임워크의 소스 코드는 2000년 가을에 open community에 릴리스되었습니다. 공동 구현 참조 사례에 따라 코드 기반의 진화에 참여함으로써 세계적 응용프로그램의 호환성과 상호 운영성이 개선 가능하게 되었습니다. 특정 코드셋에 의존하지 않는 국제화에 대한 접근 방식은 사용자가 아키텍처 고유의 코드 환경에서 작업하게

나, 급성장하고 있는 유니코드의 세계에 참여할 수 있도록 해 줍니다. Solaris 프레임워크는 다양한 코드화 및 플랫폼(예: Microsoft Windows 또는 Macintosh) 간의 상호 운영성의 보장을 위해 설계된 풍부한 데이터 변환기 모음으로 플랫폼에 구애받지 않고 확장할 수 있는 강력한 능력을 제공합니다.

또한 Solaris는 다국적 기업들의 전 세계 서버 관리 확장을 도와줍니다. 경쟁 플랫폼들과 달리 Solaris 플랫폼은 언어 서비스 관리에 서비스 기반 접근 방식을 사용합니다. 서버 관리자들은 클라이언트 시스템에 관계 없이 전 세계 네트워크를 통해 언어 서비스를 활성화시킬 수 있습니다. 클라이언트 독립적 접근 방식은 클라이언트 응용 프로그램을 변경하지 않아도 시스템을 쉽게 업그레이드 가능하게 합니다. 예를 들어, 파리의 인터넷 카페에서 보내온 전자 우편을 읽어야 하는 아랍어 사용자는 로컬 클라이언트 응용 프로그램을 수정하지 않고도 자신의 언어로 해당 전자 우편을 읽을 수 있습니다.

---

## 새로운 국제화 및 지역화 기능

다음은 Solaris 9 릴리스의 새로운 기능입니다.

- 태국, 인도, 홍콩, 터키, 이집트, 브라질, 핀란드 및 벨기에(왈론)용 추가 유니코드 (UTF-8) 로캘 지원 기능
- 유니코드 로캘의 Latin-3 문자 지원
- mp 인쇄 필터에서의 PCL 지원
- 정체 한자(홍콩) Big5+HKSCS 로캘 (zh\_HK.BIG5HK)
- 정체 한자(홍콩) UTF-8 로캘 (zh\_HK.UTF-8)
- 태국어 UTF-8 로캘 (th\_TH.UTF-8)
- 태국어 ISO8859-11 로캘 (th\_TH.ISO8859-11)
- 유니코드 로캘에서의 힌디어 스크립트 지원
- 힌디어 UTF-8 로캘 (hi\_IN.UTF-8)
- ISCII 및 UTF-8 코드화 사이의 변환을 지원하는 ISCII iconv 모듈
- 아시아 언어 Solaris용 조합 로캘
- 새 GB18030-2000 표준 지원을 위한 새로운 zh\_CN.GB18030 로캘
- HKSCS iconv 모듈
- 새로운 중국어 입력 방식
- 태국어 입력 방식 개선
- 아시아권 언어 Solaris용 입력 방식 보조 창 지원
- Fujitsu JEF, Hitachi KEIS 및 NEC JIPS용 일본어 iconv 추가 변환 모듈
- 유로화 현재 유로 화폐를 사용하고 있는 로캘은 국가 통화 심볼로 정의된 유로 화폐 기호를 포함하고 있습니다. 그 밖의 ISO8859-15 로캘들도 유로 기호, €를 지원합니다.

- 개선된 유니코드 iconv 모듈. 다양한 새 유니코드 코드화 형식과 국제적 표준 코드 세트 및 **사실상의** 업계 표준 코드 세트를 위해 iconv 모듈이 추가 및 개선되었습니다.
- 유니코드 로켈에서의 유니코드 3.1 지원.
- ISO8859-16을 위한 새 iconv 코드 변환 지원.

## 국제화 및 지역화의 개념

국제화와 지역화는 서로 차이가 있습니다. 국제화란 언어나 지역 간에 소프트웨어를 이식 가능한 상태로 만드는 과정인 반면 지역화란 소프트웨어를 특정 언어나 지역에 적응시키는 과정입니다. 국제화된 소프트웨어는, 특정 문화의 요구에 맞도록 프로그램의 동작 방식을 변경시켜주는 인터페이스를 이용해 개발될 수 있습니다. 지역화는 **로켈**이라 부르는 언어나 지역을 지원하는 온라인 정보를 구축하는 것과 관련됩니다.

모국어 및 사용자가 달라지면 완전히 재작성해야 하는 소프트웨어와 달리 국제화된 소프트웨어는 재작성이 필요없습니다. 국제화된 소프트웨어는 하나의 로켈에서 다른 로켈로 아무런 변경없이 이식할 수 있습니다. Solaris 시스템은 국제화되어 있어 국제화된 소프트웨어를 작성하기 위해 필요한 인프라와 인터페이스를 제공합니다.

## 국제화 기본 절차

국제화된 응용프로그램의 실행 가능 이미지는 언어와 지역에 따라 이식 가능합니다. 소프트웨어를 국제화하려면 다음을 수행해야 합니다:

- 이 책에서 설명하는 인터페이스를 사용하여, 재컴파일할 필요 없이 동적으로 변하는 환경을 가진 소프트웨어를 만듭니다.
- 소프트웨어를 실행 코드와 메시지로 분리합니다. 메시지는 사용자가 볼 수 있는 인쇄되거나 화면에 표시된 모든 메시지를 포함합니다. 메시지 문자열은 메시지 카탈로그로 보관합니다.

메시지 문자열을 언어나 지역에 맞추어 번역합니다. **로켈**에는 메시지 스트링과 정렬 방식을 지정하는 메소드가 포함됩니다.

지역화된 버전의 소프트웨어를 사용하려면 사용자가 특정 환경 변수를 설정해야 합니다. 그러면, 제품은 메시지를 그에 해당하는 번역된 형식으로 표시합니다. 날짜, 시간, 통화 및 기타 정보는 로켈별 규약에 따라 포맷 및 표시됩니다. 메시지 번역과 온라인 도움말 콘텐츠는 다음 도표에서 설명하는 바와 같이 서로 다른 계층에서 제공됩니다.

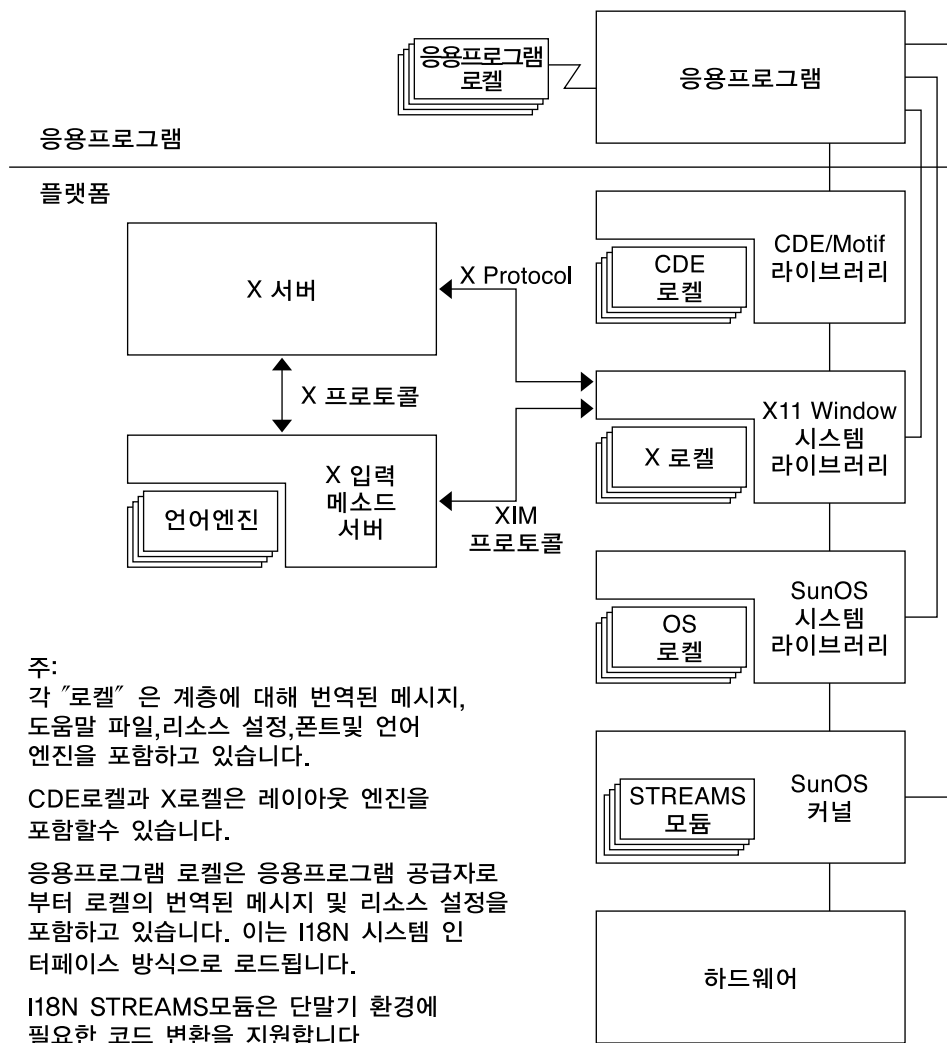


그림 1-1 Solaris 운영 환경의 로케일 기능과 구조

## Solaris 인터페이스의 지역화 기능

OS 로케일 층은 기초 로케일 데이터베이스와, 응용프로그램의 런타임에 OS 시스템 인터페이스에 연결되는 함수들을 제공합니다. 응용프로그램들은 이 OS 로케일 모듈에 표준 API를 통해 액세스합니다.

X11 로케일 계층은 X 입력 방식 및 X 출력 메소드에 인터페이스를 제공해 X11 응용프로그램이 로케일 텍스트를 입력 및 표시할 수 있게 합니다. 응용프로그램이 다양한 언어로 문자를 표시할 수 있게 하는 글꼴이 제공됩니다.



CDE/Motif는 X11 윈도우 시스템을 기반으로 설계되었습니다. 따라서 X11 API를 통해 X11 로케일을 활용할 수 있습니다. Solaris 지역화는 대상 로케일에서 데스크탑이 작동되도록 하기 위해 다양한 로케일별 CDE 응용프로그램 구성을 갖고 있습니다. 메시지 번역과 온라인 도움말 콘텐츠는 서로 다른 계층에서 제공됩니다.

---

## 로케일란?

응용프로그램의 핵심 개념은 프로그램 **로케일**의 개념과 같습니다. 로케일은 모국어 환경의 명시적 모델 및 개념입니다. 로케일의 개념은 ANSI C 언어 표준 라이브러리 정의에 명확히 정의되어 있습니다.

로케일은 국가별로 다른 형식이나 기타 사양을 갖는 여러 개의 범주로 구성됩니다. 프로그램의 로케일은 해당 로케일의 코드 세트, 날짜 및 시간 형식 규약, 통화 규약, 10진수 형식 규약 및 조합(정렬) 순서 등을 정의합니다.

로케일은 기본 언어와 그 언어가 쓰이는 나라(영역)로 이루어지며, 선택적으로 코드 세트가 포함될 수 있습니다. 대개의 경우 코드 세트가 포함됩니다. 예를 들어, 독일어는 de로서 Deutsch의 약어이고 스위스 독일어는 de\_CH로서 이 때 CH는 Confederation Helvetica의 약어입니다. 이로 인해 통화 단위 표기법과 같이 국가별로 차이를 둘 수 있습니다.

둘 이상의 로케일을 특정 언어와 연결하여 지리적 차이를 둘 수 있습니다. 예를 들어, 미국에서 영어를 사용하는 사용자는 en\_US 로케일(미국 영어)을 선택할 수 있고 영국의 영어 사용자는 en\_GB(영국 영어)를 선택할 수 있습니다.

일반적으로 로케일 이름은 LANG 환경 변수에 의해 지정됩니다. 로케일 범주는 LANG에 종속되지만 별도로 설정할 수 있는데, 이 경우 LANG은 무시됩니다. 만약 환경변수 LC\_ALL을 어떤 로케일로 설정하면, 그 변수의 값뿐만 아니라 별개의 모든 로케일 범주의 값도 그 값으로 바뀝니다.

로케일 이름 지정 규약은 다음과 같습니다:

```
language[_territory][.codeset] [@modifier]
```

두 자의 *language* 코드는 ISO 639, 두 자의 *territory* 코드는 ISO 3166의 코드이며, *codeset*은 로케일에서 사용될 코드 세트 이름이고, *modifier*는 그 수정자(modifier)가 없는 로케일에 별도로 존재하는 특성의 이름입니다.

모든 Solaris 제품 로케일은 US-ASCII 코드 값과 함께 PCS(Portable Character Set) 문자를 보존합니다.

PCS(Portable Character Set)에 대한 자세한 정보는 "X/Open CAE Specification: System Interface Definitions, Issue 5" (ISBN 1-85912-186-1)을 참조하십시오.

단일 로케일이 하나 이상의 로케일 이름을 가질 수 있습니다. 예를 들어, POSIX는 C와 동일합니다.

## 전체 및 부분 로케일

전체 Solaris 로케일은 관련 언어에서 나열된 모든 기능과 지역화된 시스템 메시지를 갖습니다. **부분 로케일**은 아무런 메시지도 설치되지 않습니다. Solaris 환경의 모든 로케일들은 관련 언어용 지역화된 메시지가 설치만 되어 있다면 지역화된 메시지를 표시할 수 있습니다. 예를 들어, 다음 로케일은 부분 또는 전체 로케일일 수 있습니다:

- de\_DE.ISO8859-1
- de\_DE.ISO8859-15
- de\_DE.UTF-8
- de\_AT.ISO8859-1
- de\_AT.ISO8859-15
- de\_CH.ISO8859-1

Language CD를 사용해 독일어 메시지 번역을 설치하면 위의 모든 로케일이 완전히 번역된 데스크탑에 액세스할 수 있어 **전체 로케일**이 됩니다. Language CD에는 다음 언어 및 로케일용 메시지 번역이 포함되어 있습니다:

- 독일어
- 프랑스어
- 스페인어
- 스웨덴어
- 이탈리아어
- 일본어
- 한국어
- 간체 한자 로케일
- 정체 한자 로케일

모든 부분 로케일은 소프트웨어 CD에 포함되어 있습니다. 메시지 번역은 Language CD에 포함되어 있습니다.

모든 영어 로케일은 전체 로케일이며 소프트웨어 CD에 포함되어 있습니다.

## 로케일의 영향을 받는 동작

문화에 따라 날짜 및 시간 기록, 숫자 형식, 단어와 구 구분 및 자료 인용 등을 위한 규약이 달라집니다. 시스템 전체적으로 로케일은 다음 항목의 동작을 결정합니다:

- 텍스트 데이터의 코드화 및 처리
- 리소스 파일의 언어와 인코딩 방식 결정
- 텍스트 문자열의 렌더링과 레이아웃
- 클라이언트간 텍스트 통신용으로 사용된 텍스트 교환하기
- 입력 방식(생성되는 코드 세트)의 선택과 텍스트 데이터의 처리
- 문화적으로 다른 글꼴과 아이콘 파일
- 활동 및 파일 유형
- 사용자 인터페이스 정의(UI) 파일
- 날짜 및 시간 형식
- 숫자 형식

- 통화 형식
- 조합 순서
- 로케일에 따른 정규식 처리
- 정보 제공 메시지 및 진단 메시지와 대화식 응답용 형식

Solaris 환경은 언어와 문화에 따른 정보를 응용프로그램에서 분리하여 응용프로그램 밖에 저장합니다. 이렇게 함으로써 각 시장에 맞추어 응용프로그램을 번역, 재작성 및 재컴파일할 필요성을 제거합니다. 새로운 시장에 진출할 때 필요한 것은 외부 정보를 현지 언어 및 고객에 맞게 지역화하는 것 뿐입니다.

## 로케일 범주

로케일 범주는 다음과 같습니다:

LC_CTYPE	문자 처리 기능의 동작을 제어합니다.
LC_TIME	달과 요일 이름, 공통된 전체 표현 및 약식 표현을 포함한 날짜와 시간 형식을 지정합니다.
LC_MONETARY	로케일용 통화 기호, 1000 구분자, 기호 위치, 분수 자리수 등을 포함한 통화 형식을 지정합니다.
LC_NUMERIC	10진수 분리자(또는 기수 문자), 1000 단위 구분자 및 그룹화를 지정합니다.
LC_COLLATE	조합 순서 및 로케일용 정규 표현 정의를 지정합니다.
LC_MESSAGES	지역화된 메시지를 표시하는 언어, 로케일의 긍정과 부정 응답(yes 및 no 문자열과 표현)을 지정합니다.
LO_LTYPE	언어 렌더링에 관한 정보를 제공하는 레이아웃 엔진을 지정합니다. 언어 렌더링(또는 텍스트 렌더링)은 텍스트 모양과 방향으로 구성됩니다.

---

## 지역화용 로케일 범주 사용

제품의 지역화는 대상 언어나 지역의 원 사용자의 자문을 얻어 수행해야 합니다. 특정 정보 유형과 형식은 개발자에겐 완전히 명백하고 보편적인 것으로 보일 수 있지만 사용자에게엔 모호하고 잘못되었거나 심지어 불쾌하게까지 느껴질 수 있습니다. 다음 절에서는 제품을 성공적으로 지역화할 수 있도록 제어 및 지정할 수 있는 Solaris 운영 환경의 요소들을 설명합니다.

## 시간 형식

다음 표는 개별 로캘이 11:59 P.M.을 쓰는 방식 중 일부를 보여줍니다.

표 1-1 국제 시간 형식

로캘	형식
캐나다어	23:59
핀란드어	23.59
독일어	23.59 Uhr
노르웨이어	23.59
태국어	23:59
영국	23:59

시간은 12 시간 시계 및 24 시간 시계로 표현됩니다. 시간 및 분 구분자는 콜론(:) 또는 마침표(.)를 사용할 수 있습니다.

시간대 구분은 국가별로 이루어집니다. 시간대는 협정 세계시인 UTC (또는 그리니치 표준시인 GMT)에서 몇 시간 앞인지, 또는 뒤인지에 따라 설명할 수 있지만 이 숫자는 항상 정수인 것은 아닙니다. 예를 들어, 뉴펀들랜드는 인접 시간대와 30분 차이가 있는 시간대입니다.

일광 절약 시간(DST)은 국가별로 서로 다른 날짜에 시작하거나 끝납니다. 많은 국가들은 아예 DST를 설정하지 않습니다. 아울러 일광 절약 시간은 시간대 안에서도 다를 수 있습니다. 미국에서는 주 정부가 이를 결정합니다.

## 날짜 형식

다음 표는 전 세계적으로 사용되는 날짜 형식 중 일부를 보여줍니다. 한 국가 안에서도 날짜 형식에 차이가 있을 수 있다는 점에 주목하십시오.

표 1-2 국제 날짜 형식

로캘	규약	예
캐나다(영어)	dd/mm/yy	24/08/01
덴마크어	yyyy-mm-dd	2001-08-24
핀란드어	dd.mm.yyyy	24.08.2001
프랑스어	dd/mm/yyyy	24/08/2001
독일어	yyyy-mm-dd	2001-08-24

표 1-2 국제 날짜 형식 (계속)

로케일	규약	예
이탈리아어	dd/mm/yy	24/08/01
노르웨이어	dd-mm-yy	24-08-01
스페인어	dd-mm-yy	24-08-01
스웨덴어	yyyy-mm-dd	2001-08-24
영국	dd/mm/yy	24/08/01
미국	mm-dd-yy	08-24-01
태국어	dd/mm/yyyy	24/08/2001

## 숫자

영국과 미국은 소수 자릿수를 마침표를 사용해 표시하는 몇 안 되는 국가들 중 두 국가입니다. 그 밖의 많은 국가들은 마침표 대신 콤마를 사용합니다. 소수 구분 기호는 **기수** 문자라고도 부릅니다. 마찬가지로 영국과 미국은 콤마를 사용해 1000 단위 그룹을 구분하지만 다른 많은 국가들은 대신 마침표를 사용하고 일부 국가는 1000 단위를 좁은 공백으로 구분합니다.

로케일 지향 형식을 포함한 데이터 파일은 다른 로케일의 시스템으로 전달될 때 잘못 해석되는 경우가 종종 발생합니다. 예를 들어, 프랑스어 형식의 숫자를 포함하고 있는 파일은 영국 프로그램에는 적합하지 않습니다.

다음 표는 흔히 사용되는 숫자 형식 일부를 보여줍니다.

표 1-3 국제 숫자 규약

로케일	큰 숫자
캐나다(영어)	4,294,967.00
덴마크	4.294 967.295,00
핀란드	4 294 967 295,00
프랑스어	4 294 967 295,00
독일어	4,294,967.00
이탈리아어	4.294.967,00
노르웨이어	4.294.967.295,00
스페인어	4.294.967.295,00
스웨덴어	4 294 967 295,00

표 1-3 국제 숫자 규약 (계속)

로케	큰 숫자
영국	4,294,967,295.00
미국	4,294,967,295.00
태국어	4,294,967,295.00

주 - 목록의 숫자를 구분하는 방법을 지정하는 특별한 로케 규약은 없습니다.

## 통화

통화 단위 및 표현 순서는 전 세계적으로 큰 차이가 있습니다. 현지 통화 기호와 국제적으로 통용되는 통화 기호가 서로 다를 수 있습니다. 다음 표는 일부 국가의 통화 형식을 보여줍니다.

표 1-4 국제 통화 규약

로케	통화	예제
캐나다(영어)	달러 (\$)	\$1,234.56
캐나다(프랑스어)	달러 (\$)	1 234,56\$
덴마크어	크로네(kr)	Kr 1.234,56
핀란드어	유로(€)	€ 1 234,56
프랑스어	유로(€)	€ 1,234
일본어	엔(¥)	¥ 1,234
노르웨이어	크로네(kr)	kr 1.234,56
스웨덴어	크로너(Kr)	1 234,56 Kr
영국	파운드 (£)	£1,234.56
미국	달러 (\$)	\$1,234.56
태국어	바트	2539 바트
유로	유로(€)	€ 5,000

Solaris 9 소프트웨어는 유로 통화를 지원합니다. 하지만 역 호환성을 위해 현지 통화 기호도 사용할 수 있습니다.

표 1-5 유로 통화를 지원하는 사용자 로케일

지역	로케일 이름	ISO 코드 세트
오스트리아	de_AT.ISO8859-15	8859-15
벨기에(프랑스어)	fr_BE.ISO8859-15	8859-15
벨기에(플란더즈어)	nl_BE.ISO8859-15	8859-15
덴마크	da_DK.ISO8859-15	8859-15
핀란드	fi_FI.ISO8859-15	8859-15
프랑스	fr_FR.ISO8859-15	8859-15
독일	de_DE.ISO8859-15	8859-15
아일랜드	en_IE.ISO8859-15	8859-15
이탈리아	it_IT.ISO8859-15	8859-15
네덜란드	nl_NL.ISO8859-15	8859-15
포르투갈	pt_PT.ISO8859-15	8859-15
카탈로니아어 스페인	ca_ES.ISO8859-15	8859-15
에스토니아	et_EE.ISO8859-15	8859-15
스페인	es_ES.ISO8859-15	8859-15
스웨덴	sv_SE.ISO8859-15	8859-15
영국	en_GB.ISO8859-15	8859-15
미국	en_US.ISO8859-15	8859-15

유로 로케일은 ISO8859-15 코드 세트를 기본으로 합니다.

**변환된** 통화량은 원래의 양보다 많거나 적은 공간을 차지할 수 있다는 점에 주의하십시오. 예: \$1,000는 € 1.307.000이 될 수 있습니다.

현재 유로 사용 지역 내의 로케일용 로케일 설정 상태 예는 로케일 유틸리티의 LC\_MONETARY 피연산자에 대한 것입니다. 예를 들어, 독일의 상태는 다음 표에 표시되어 있습니다.

표 1-6 독일어 로케일과 관련 LC\_MONETARY

로케일	LC_MONETARY
de_DE.ISO8859-1	DM
de_DE.ISO8859-15	유로
de_DE.UTF-8	유로

표 1-6 독일어 로케일과 관련 LC\_MONETARY (계속)

로케일	LC_MONETARY
de_DE.ISO8859-15@euro	유로
de_DE.UTF-8@euro	유로

## 언어 단어 및 글자 차이

이 절에서는 언어간의 중요한 차이점을 설명합니다.

### 단어 분리자

영어에서 단어는 보통 공백 문자에 의해 구분됩니다. 하지만 중국어, 일본어 및 태국어와 같은 언어에서는 단어 사이에 분리자가 없는 경우가 많습니다.

### 정렬 순서

특정 문자용 정렬 순서는 언어별로 차이가 있습니다. 예를 들어, 문자 "ö"는 독일에서 보통의 "o"와 같이 분류됩니다. 하지만 스웨덴에서는 별개의 문자로 분류되며 알파벳의 끝 문자에 해당합니다. 일부 언어에서 문자에는 문자 시퀀스의 우선 순위를 결정하기 위해 가중치가 주어집니다. 예를 들어, 태국어 사전은 서로 가중치가 다른 문자들의 순서를 통해 정렬을 규정합니다.

### 문자 세트

문자 세트는 영문자와 특수 문자에서 서로 다를 수 있습니다. 영문자에는 26개 문자가 포함되지만 일부 언어에는 훨씬 많은 문자가 포함됩니다. 예를 들어, 일본어는 20,000자 이상이 포함될 수 있으며 중국어에는 이보다도 더 많은 문자가 포함될 수 있습니다.

### 서유럽 문자

대부분의 서유럽 국가들의 문자는 영어를 사용하는 국가들에서 사용되는 표준 26개 문자와 유사하지만 때로는 일부 기본 문자와 기호가 표시된(또는 악센트 부호가 있는) 문자 및 일부 연결 문자가 추가됩니다.

### 일본어 텍스트

일본어 텍스트는 세 가지 스크립트가 혼합되어 구성됩니다: 한자로부터 발생된 간지 표의 문자 및 두 가지 표음 문자(또는 음절 문자표)인 히라가나와 가타카나.



히라가나의 각 문자는 가타카나에도 상응하는 문자가 있지만 히라가나는 가장 공통된 문자이고 블록체보다는 초서체에 가깝습니다. 간지 문자는 핵심 단어를 쓰는데 사용됩니다. 가타카나는 대개 “외래어”, 다시 말해 일본어 이외의 언어에서 “수입된” 단어를 표현하는데 사용됩니다.

간지는 수 만개의 문자를 갖고 있지만 시간이 지날수록 자주 사용되는 문자 수는 줄어들고 있습니다. 평균적인 일본어 작가들은 대략 2000자의 간지 문자를 사용할 수 있는 어휘력을 갖고 있지만 지금은 약 3500자만이 자주 사용되고 있습니다. 하지만 컴퓨터 시스템은 일본 공업 표준(JIS)이 요구하는 7000자 이상을 지원해야 합니다. 아울러 약 170개의 히라가나와 가타카나 문자가 있습니다. 평균적으로 일본어 텍스트의 55%가 히라가나이며 35%는 간지 그리고 10%가 가타카나입니다. 아라비아 숫자와 로마 문자 역시 일본어 텍스트에 사용됩니다.

간지를 전혀 사용하지 않는 것도 가능하지만 대부분의 일본 독자들은 간지를 전혀 사용하지 않고 작성한 텍스트는 이해하기 힘들어 합니다.

## 한국어 텍스트

한국어 텍스트는 한글이라 불리는 표음 문자 체계를 사용해 쓸 수 있습니다. 한글은 자모로 알려져 있는 자음과 모음으로 구성된 11,000자 이상의 문자를 갖고 있습니다. 전체 한글 문자 사전 중에서 대략 3000자가 한국어 컴퓨터 시스템에서 일상적으로 사용됩니다. 한국인들은 중국에서 발명된 문자를 바탕으로 한 한자라는 표의 문자도 사용합니다. 한국어 텍스트에는 6,000자 이상의 한자 문자가 필요합니다. 한자는 대개 한글의 의미가 모호할 때 혼동을 방지하기 위해 사용됩니다. 한글 문자는 자음과 모음을 결합하여 형성됩니다. 결합한 뒤에는 한 개의 음절을 구성할 수 있는데 이것이 한글 문자입니다. 한글 문자는 보통 사각형으로 배열되며 이 그룹은 한자 문자와 동일한 공간을 점유합니다. 아라비아 숫자, 로마 문자 및 특수 기호 문자도 한국어 텍스트에 사용됩니다.

## 태국어 텍스트

태국어 문자는 디스플레이 화면에 네 개의 디스플레이 셀로 구성되는 열 위치로 정의할 수 있습니다. 각 열 위치에 최대 세 개의 문자를 표현할 수 있습니다. 디스플레이 셀의 구성은 태국어 문자의 분류법에 따릅니다. 일부 태국어 문자는 다른 문자의 분류법과 함께 구성될 수 있습니다. 함께 구성할 수 있는 경우에는 두 문자가 동일한 셀에 표시됩니다. 그렇지 않으면 별개의 셀들에 표시됩니다.

## 중국어 텍스트

중국어는 보통 hanzi라는 표의 문자로만 구성됩니다.

- 중국(PRC)에는 유니코드 3.0에 정의되어 있는 모든 CJK 확장 A 문자를 포함해 자주 사용되는 대략 7000자의 GB2312 (zh 로캘) hanzi 문자, GBK 문자 세트(zh.GBK 로캘) 20,000개 문자 그리고 GB18030-2000 문자 세트(zh\_CN.GB18030 로캘) 약 30,000개 문자가 있습니다.
- 대만에서 가장 많이 사용되는 문자 세트는 CNS11643-1992 (zh\_TW 로캘) 및 Big5 (zh\_TW.BIG5 로캘)입니다. 이들은 약 13,000자의 hanzi 문자를 공유합니다.

- 홍콩에서는 4702개 문자가 Big5 문자 세트에 추가된 Big5-HKSCS 문자 세트 (zh\_HK.BIG5HK)가 사용됩니다.

특정 문자가 기본 문자가 아닌 경우에는 대개 두 개 이상의 부분으로 구성되며 이 중 두 자는 매우 자주 쓰이는 문자입니다. 두 부분으로 구성된 문자에서 한 부분은 보통 의미를, 다른 한 부분은 발음을 나타냅니다. 두 부분이 모두 의미를 나타내는 경우도 있습니다. 부수는 가장 중요한 요소이며 문자는 전통적으로 부수별로 정렬되며 수백개의 부수 문자가 있습니다. 하나의 음은 서로 상호 교환해 사용할 수 없는 여러 개의 서로 다른 문자들로 표현될 수 있습니다. 하나의 문자는 여러 가지 음을 가질 수 있습니다.

일부 문자가 문맥에 따라 다른 문자보다 적합할 수 있습니다—어떤 문자가 적합한지는 성조의 사용에 의해 발음에 따라 구분됩니다. 이와 대조적으로 일본어와 한국어는 성조가 많지 않습니다.

중국어 표현하는 음성학적 체계가 몇 개 있습니다. 중국에서 가장 흔한 것은 *pinyin* 으로서 로마 문자를 사용하며 Beijing과 같은 장소 이름을 표기하기 위해 서구에서 널리 사용되고 있습니다. Wade-Giles 체계는 이보다 오래된 음성학적 체계로서 과거에 Peking과 같이 장소 이름에 사용되었습니다. 대만에서는 대신 *zhuyin* (또는 *bopomofo*) 이라는 고유 문자 형태를 갖고 있는 표음 문자가 자주 사용됩니다.

## 히브리어 텍스트

히브리어 텍스트는 히브리어와 이디시 언어에서 글을 쓰기 위해 사용되며 영어보다 수 천년 앞서 생겼습니다. 히브리어는 글자는 오른쪽에서 왼쪽으로 쓰고 읽고, 숫자는 왼쪽에서 오른쪽으로 읽는다는 점에서 양방향 문자에 속합니다. 히브리어 텍스트에 포함된 모든 영어 텍스트 역시 왼쪽에서 오른쪽으로 읽습니다.

히브리어는 27개 문자를 사용하며 표준 라틴(또는 영어) 문자의 구두점과 숫자를 사용합니다. 히브리어 텍스트에는 또한 모음과 발음 기호가 포함되어 있습니다. 이 기호들은 기본 문자 안의 점(Dagesh), 문자 아래의 모음 기호 또는 문자 상단 왼쪽의 약센트 기호로 표시됩니다. 이 기호들은 보통 예배용 텍스트에서만 사용되며 일상적으로는 거의 사용되지 않습니다. 히브리어에서는 대문자를 사용하지 않습니다.

## 힌디어 텍스트

힌디어 텍스트는 "신의 기록"이라는 의미를 갖고 있는 Devanagari라 부르는 문자로 작성됩니다. 힌디어는 표음 언어로서 연속적인 음절로 쓰여집니다. 각각의 음절은 다음 세 가지 문자 조각(Devanagari 문자)으로 구성됩니다: 자음 문자, 독립 모음 및 종속 모음 기호. 음절 자체는 자음과 기본 모음으로 구성되며 별도로 전치 모음이 있을 수 있습니다. 기준선에서 시작되는 영어와 달리 Devanagari 문자는 문자의 상단에 쓰여 있는 가로선(head stroke)에 매달려 있습니다. 이 문자들은 문맥에 따라 결합하거나 모양을 변경할 수 있습니다. 히브리어와 마찬가지로 힌디어 텍스트는 대소문자 구분을 하지 않습니다.

---

## 키보드 차이점

미국 키보드상의 모든 문자가 다른 키보드에 나타나지는 않습니다. 마찬가지로 다른 키보드에는 미국 키보드에서 볼 수 없는 많은 문자들이 포함되곤 합니다.

---

주 - SPARC™ 시스템에서는 Compose 키를 사용해 지원되는 ISO8859 문자 세트의 분음 부호를 갖고 있는 라틴 문자를 만들 수 있습니다.

Compose 키는 라틴 기반 로케일과 사용할 수 있지만 UTF-8 로케일을 제외한 한국어, 중국어 또는 일본어 로케일과는 사용할 수 없습니다.

입력은 Solaris 운영 환경에 의해 처리되기 때문에 어떤 로케일의 입력 문자든 임의의 키보드를 사용해 입력할 수 있습니다.

---

---

## 용지 크기의 차이

각 국가별로 일반적으로 사용되는 용지 크기들이 몇 개 있습니다. 일반적으로 이 크기 중 한 가지가 다른 것들보다 훨씬 일반적입니다. 대부분의 국가들은 ISO 표준 216을 준수합니다: "Writing paper and certain classes of printed matter-Trimmed sizes-A and B series."

국제화된 응용프로그램들은 이용 가능한 페이지 크기에 대해 가정해선 안 됩니다. Solaris 시스템은 출력 페이지 크기 추적을 지원하지 않습니다. 페이지 크기 추적은 응용프로그램의 책임입니다. 다음 표는 공통적인 국제 페이지 크기를 보여줍니다.

표 1-7 공통 국제 페이지 크기

용지 유형	규격	국가
ISO A4	21.0 cm X 29.7 cm	미국을 제외한 모든 국가
ISO A5	14.8 cm X 21.0 cm	미국을 제외한 모든 국가
JIS B4	25.9 cm X 36.65 cm	일본
JIS B5	18.36 cm X 25.9 cm	일본
U.S. Letter	8.5 인치 X 11 인치	미국 및 캐나다
U.S. Legal	8.5 인치 X 14 인치	미국 및 캐나다



# 일반 국제화 기능

---

이 절은 Solaris 9 환경에 포함된 여러 국제화 기능에 대해 다룹니다.

- “코드 세트 독립 지원” 37 페이지
- “로캘 데이터베이스” 39 페이지
- “프로세스 코드 형식” 40 페이지
- “멀티바이트 지원 환경” 40 페이지
- “동적으로 링크된 응용프로그램” 41 페이지
- “변경된 인터페이스” 41 페이지
- “ctype 매크로” 42 페이지
- “libc의 국제화 API” 43 페이지
- “genmsg 유틸리티” 50 페이지

---

## 코드 세트 독립 지원

EUC는 Extended UNIX Code의 축약입니다. Solaris 9 운영 환경은 일본의 PC-Kanji (Shift\_JIS로 더 잘 알려짐), 대만의 Big5, 중화인민공화국의 GBK와 같은 비EUC 인코딩을 지원합니다. 대규모 컴퓨터 시장이 비EUC 코드 세트 지원을 요구하기 때문에 Solaris 9 환경은 EUC와 비EUC 코드 세트 지원을 모두 가능하게 하는 견고한 프레임워크를 제공합니다. 이 지원을 **코드 세트 독립** 또는 CSI라고 합니다.

CSI의 목적은 Solaris 운영 환경 라이브러리와 명령으로부터, 특정한 코드 세트나 인코딩 메소드에 대한 의존성을 제거하는 데 있습니다. CSI 아키텍처는 Solaris 운영 환경이 어떤 종류의 UNIX 파일 시스템에서도 사용할 수 있는 인코딩 방식을 지원하도록 해 줍니다. CSI는 UTF-8, PC-Kanji 및 Big5와 같은 많은 새 코드 세트를 지원합니다.

## CSI 접근 방법

CSI는 응용프로그램 및 플랫폼 소프트웨어 개발자가 UTF-8 같은 모든 인코딩과 무관하게 코드를 작성할 수 있도록 해 주며, 또한 소스 코드를 수정하지 않고서도 새로운 인코딩 방식을 적용할 수 있도록 해 줍니다. 이러한 아키텍처는 Java의 경우 응용프로그램이 UTF-16에 의존적이라는 점에서 Java™ 국제화와는 다른 접근 방식을 갖습니다.

많은 기존 국제화된 응용프로그램(예를 들어, Motif)이 기본 시스템에서 자동으로 CSI 지원을 상속합니다. 이러한 응용프로그램은 수정 없이 새 로케에서 작동합니다.

CSI는 본질적으로 모든 코드 세트로부터 독립적입니다. 그러나 파일 코드 인코딩(코드 세트)에 대한 다음 가정이 Solaris 9 환경에 적용됩니다.

- 파일 코드는 ASCII의 슈퍼 세트입니다.
- 널 바이트 값(0x00)은 널로 종결되는 멀티바이트 문자열의 지원을 위한 멀티바이트 문자 바이트의 일부로 나타나지 않습니다.
- ASCII 슬래시 문자 바이트 값(0x2f)은 UNIX 경로 이름의 지원을 위한 멀티바이트 문자 바이트의 일부로 나타나지 않습니다.

## CSI화 된 명령

이 절은 Solaris 9 환경의 CSI화 된 명령을 나열합니다. 각 명령어의 설명서 페이지에는 명령이 CSI화 되었는지를 나타내는 속성 부분이 있습니다.

모든 명령은 특별히 표시되지 않는 한, /usr/bin 디렉토리에 있습니다.

/usr/lib/diffh	/usr/xpg4/bin/nohup	chown
/usr/sbin/accept	/usr/xpg4/bin/od	cmp
/usr/sbin/reject	/usr/xpg4/bin/pr	col
/usr/ucb/lpr	/usr/xpg4/bin/rm	comm
/usr/xpg4/bin/awk	/usr/xpg4/bin/sed	compress
/usr/xpg4/bin/cp	/usr/xpg4/bin/sort	cpio
/usr/xpg4/bin/date	/usr/xpg4/bin/tail	csh
/usr/xpg4/bin/du	/usr/xpg4/bin/tr	csplit
/usr/xpg4/bin/ed	/usr/xpg4/bin/vedit	cut
/usr/xpg4/bin/edit	/usr/xpg4/bin/vi	diff
/usr/xpg4/bin/egrep	/usr/xpg4/bin/view	diff3
/usr/xpg4/bin/env	acctcom	disable
/usr/xpg4/bin/ex	apropos	echo
/usr/xpg4/bin/expr	batch	expand
/usr/xpg4/bin/fgrep	bdiff	file
/usr/xpg4/bin/lp	cancel	find
/usr/xpg4/bin/ls	cat	fold
/usr/xpg4/bin/more	catman	ftp
/usr/xpg4/bin/mv	chgrp	gencat
/usr/xpg4/bin/nice	chmod	geteopt

getoptcv	printf	strings
head	priocntl	sum
join	ps	tabs
jsh	pwd	tar
kill	rcp	tee
ksh	red	touch
lp	remsh	tty
man	rksh	uncompress
mkdir	rsmdir	unexpand
msgfmt	rsh	uniq
news	script	unpack
nroff	sdiff	wc
pack	settime	whatis
paste	sh	write
pcat	split	xargs
pg	strconf	zcat

## Solaris 9 CSI화 된 라이브러리

libc(/usr/lib/libc.so)의 거의 모든 함수가 CSI화 되었습니다. 그러나 libc의 다음 함수는 EUC 종속 함수이기 때문에 CSI화 되지 않았습니다.

- csetcol()
- csetlen()
- euccol()
- euclen()
- eucscol()
- getwidth()
- csetno()
- wcsetno()

Solaris 9 제품에서 libgen /usr/ccs/lib/libgen.a 및 libcurses /usr/ccs/lib/libcurses.a 는 국제화되었지만 CSI화 되지 않았습니다.

---

## 로켈 데이터베이스

로켈 데이터베이스는 Solaris 운영환경 내부적인 것이며, 이후 릴리스에서 변경될 수 있습니다. 그러므로 국제화된 응용프로그램을 개발할 때는, 로켈 데이터베이스를 직접 액세스하지 마십시오. 대신, "libc의 국제화 API" 43 페이지에서 설명된 libc의 국제화 API를 사용하십시오.

---

주 - Solaris 9 환경으로 작업할 때 Solaris 9 제품에 포함된 로켈 데이터베이스를 사용하십시오. 이전 Solaris 버전의 로켈을 사용하지 마십시오.

---

---

## 프로세스 코드 형식

Solaris 9 제품에서 'wide-character'로 알려져 있는 프로세스 코드 형식은 Solaris 운영환경 내부적인 것이며 향후 릴리스에서 변경될 것입니다. 따라서, 국제화된 응용프로그램을 개발할 때 프로세스 코드 형식이 동일하다고 가정하지 마십시오. 대신 "libc의 국제화 API" 43 페이지에서 설명된 libc의 국제화 API를 사용하십시오.

---

주 - 모든 유니코드의 프로세스 코드는 UTF-32 표현에 있습니다. UTF-32에 대한 자세한 내용은 The Unicode Consortium의 "Unicode Standard Annex #19: UTF 32" 및 "Unicode Standard Annex #27: Unicode 3.1", 또는 <http://www.unicode.org/>를 참조하십시오.

---

---

## 멀티바이트 지원 환경

중국어, 일본어 또는 한국어 문자와 같은 멀티바이트 문자는 단일 바이트로 저장할 수 없는 문자입니다. 이러한 문자는 기억 장치의 2, 3 또는 4바이트가 필요합니다. 더 정확한 정의는 ISO/IEC 9899:1990 부속 절 3.13에서 찾을 수 있습니다.

추가된 새 국제화 기능, ISO/IEC 9899:1990라고도 하는 ANSIC에 대한 개정 1은 일괄하여 멀티바이트 지원 환경(MSE)이라고도 합니다. 개정 1은 상태가 있는 멀티바이트 코드 세트와 더 나은 와이드캐릭터 처리 지원을 위한 추가 국제화 API를 정의합니다.

프로그래밍 모델은 멀티바이트 문자가 논리 장치에서 읽히고 내부적으로 넓은 문자로 저장 가능하게 합니다. 넓은 문자는 프로그램에 의해 그 자체로 논리적 엔티티로 처리될 수 있습니다. 최종적으로 이 넓은 문자는 적절한 변환을 거쳐 논리적 단위로 외부장치로 쓰여질 수 있습니다.

절차는 단일바이트 문자가 읽히고 조작되며 재작성되는 방식과 유사합니다. MSE는 프로그램이 단일바이트 문자에 사용되는 동일한 프로그래밍 모델을 사용하여 멀티바이트 문자를 처리하도록 작성 가능하게 합니다.



---

## 동적으로 링크된 응용프로그램

Solaris 9 제품 사용자는 동적 연결 또는 정적 연결을 사용하여 libc와 같은 시스템 라이브러리와 응용프로그램을 연결하는 방법을 선택할 수 있습니다. 시스템 라이브러리에 국제화 기능을 필요로 하는 응용프로그램은 동적으로 링크되어야 합니다. 만약 응용프로그램이 정적으로 링크가 되어 있을 경우, setlocale을 호출해서 로케일을 C나 POSIX 이외의 다른 로케일로 설정할 수 없습니다. 정적으로 링크된 응용프로그램은 C 및 POSIX 로케일에서만 작동할 수 있습니다.

기본적으로, 링커 프로그램은 응용프로그램을 동적으로 연결하려고 시도합니다. 링커 및 컴파일러에 대한 명령줄 옵션이 -Bstatic 또는 -dn 사양을 포함하면 응용프로그램이 정적으로 링크될 수 있습니다. 기존 응용프로그램이 /usr/bin/ldd 명령을 사용하여 동적으로 링크되었는지 확인할 수 있습니다.

예를 들어, 다음을 입력하면

```
% /usr/bin/ldd /sbin/sh
```

명령은 /sbin/sh 명령이 다음 응답에서 볼 수 있듯이 동적으로 링크된 프로그램이 아님을 나타냅니다.

```
ldd: /sbin/sh: file is not a dynamic executable or shared object
```

다음을 입력하면

```
% /usr/bin/ldd /usr/bin/ls
```

명령이 다음 메시지를 표시합니다.

```
libc.so.1 => /usr/lib/libc.so.1
libdl.so.1 => /usr/lib/libdl.so.1
```

이 메시지는 /usr/bin/ls 명령이 두 라이브러리, libc.so.1 및 libdl.so.1과 동적으로 링크되었음을 나타냅니다.

---

## 변경된 인터페이스

libw 및 libintl은 libc로 이동하여 더 이상 libw 및 libintl에 없습니다.

공유 객체(shared object)는 기존 응용프로그램에 대한 런타임 호환성을 보장하고, 아카이브와 함께 응용프로그램을 빌드하기 위한 컴파일 환경의 호환성도 제공합니다. 그러나 더 이상 libw나 libintl에 프로그램을 링크시키지 말아야 합니다.

필터에 대한 자세한 내용은 *Linker and Libraries Guide*를 참조하십시오.

다음 목록은 libw의 스택 시작점을 보여줍니다.

fgetc	iswpunct	wscncat	wscoll
fgetws	iswspace	wcsncmp	wscopy
fputc	iswupper	wcsncpy	wscspn
fputws	iswxdigit	wcspbrk	wsdup
getc	putc	wcsrchr	wslen
getwchar	putwchar	wcsspn	wsncasecmp
getws	putws	wcstod	wsncat
isenglish	strtows	wcstok	wsncmp
isideogram	towlower	wcstol	wsncpy
isnumber	towupper	wcstoul	wspbrk
isphonogram	ungetc	wcswcs	wsprintf
isspecial	watoll	wcswidth	wsrchr
iswalnum	wscat	wcsxfrm	wsscancf
iswalpha	wcschr	wctype	wsspn
iswcntrl	wcscmp	wcwidth	wstod
iswctype	wscoll	wscasecmp	wstok
iswdigit	wscopy	wscat	wstol
iswgraph	wscspn	wchr	wstoll
iswlower	wcsftime	wscmp	wstostr
iswprint	wcslen	wscol	wsxfrm

이 보다 짧은 다음 목록은 libintl의 스테브 시작점을 나열합니다.

```
bindtextdomain
dcgettext
dgettext
gettext
textdomain
```

---

## ctype 매크로

문자 분류 및 문자 변환 매크로는 `/usr/include/ctype.h` 에서 정의됩니다. Solaris 9 환경은 XPG4가 정의하는 문자 분류 및 변환 의미를 지원하는 ctype 매크로 세트를 제공합니다. 모든 XPG4 및 XPG4.2 응용프로그램이 자동으로 새 매크로에 액세스하려면 다음 조건 중 하나를 충족해야 합니다.

- `_XPG4_CHAR_CLASS`가 정의됩니다.
- `_XOPEN_SOURCE` 및 `_XOPEN_VERSION=4` 가 정의됩니다.
- `_XOPEN_SOURCE` 및 `_XOPEN_SOURCE_EXTENDED=1` 가 정의됩니다.

`_XOPEN_SOURCE`, `_XOPEN_VERSION` 및 `_XOPEN_SOURCE_EXTENDED`가 새 ctype 매크로 외에 추가 XPG4 관련 기능을 가져오기 때문에 비XPG4 또는 XPG4.2 응용프로그램은 `__XPG4_CHAR_CLASS__`를 사용해야 합니다.

해당 ctype 함수도 존재합니다. Solaris 9 환경 함수는 XPG4 의미도 지원합니다. 자세한 내용은 ctype(3C) 설명서 페이지를 참조하십시오.

---

## libc의 국제화 API

Solaris 9 환경은 두 개의 API 세트를 제공합니다.

- 멀티바이트(파일 코드)
- 넓은 문자(프로세스 코드)

와이드캐릭터는 논리적 엔티티로서 고정폭의 단위입니다. 따라서, 멀티바이트 문자를 사용할 때처럼 계속해서 문자의 경계를 추적해야 할 필요가 없습니다.

프로그램이 파일로부터 입력을 받을 때 fscanff(3S) 및 fwscanf(3S)와 같은 입력 함수로 직접 파일의 멀티바이트 데이터를 넓은 문자 프로세스 코드로 변환하거나 입력 후 mbtowc(3C) 및 mbsrtowcs(3C)를 사용하여 변환할 수 있습니다. 넓은 문자 형식에서 멀티바이트 문자 형식으로 출력 데이터를 변환하려면 fwprintf(3S) 및 fprintf(3S)와 같은 출력 함수를 사용하거나 출력 후 wctomb(3C) 및 wcsrtombs(3C)와 같은 변환 함수를 적용하십시오.

이 장의 나머지 표는 Solaris 9 제품에 포함된 국제화 API에 대해 설명합니다.

다음 표는 libc의 메시지 처리 함수 API를 설명합니다.

표 2-1 libc의 메시지 처리 함수

라이브러리 루틴	설명
catclose()	메시지 카탈로그 닫기
catgets()	프로그램 메시지 읽기
catopen()	메시지 카탈로그 열기
dgettext()	도메인을 지정하여 메시지 카탈로그에서 메시지 얻기
dcgettext()	도메인과 범주를 지정하여 메시지 카탈로그에서 메시지 얻기
textdomain()	현재 도메인을 설정 및 질의
bindtextdomain()	메시지 도메인에 대한 경로 바인드
gettext()	메시지 데이터베이스에서 텍스트 문자열 검색

다음 표는 libc의 코드 변환 기능 API를 설명합니다.

표 2-2 libc의 코드 변환

라이브러리 루틴	설명
iconv()	코드 변환
iconv_close()	변환 설명자 할당 해제
iconv_open()	변환 설명자 할당

다음 표는 libc의 정규 표현식 API를 설명합니다.

표 2-3 libc의 정규 표현식

라이브러리 루틴	설명
regcomp()	정규식 컴파일
regexec()	정규식 매칭 실행
regerror()	오류 코드에서 오류 메시지로의 매핑을 제공
regfree()	regcomp()가 할당한 사용 가능 메모리
fnmatch()	파일 이름 또는 경로 이름 일치

다음 표는 libc의 넓은 문자 함수 API를 설명합니다.

표 2-4 libc의 넓은 문자 클래스

라이브러리 루틴	설명
wctype()	문자 클래스 정의
wctrans()	문자 매핑 정의

다음은 libc의 로캘 수정 및 질의를 나열합니다.

표 2-5 libc의 로캘 수정 및 질의

라이브러리 루틴	설명
setlocale()	프로그램의 로캘 수정 및 질의

다음 표는 libc의 로캘 데이터 질의를 나열합니다.

표 2-6 libc의 로캘 데이터 질의

라이브러리 루틴	설명
nl_langinfo()	현재 로캘의 언어 및 문화 정보 얻기
localeconv()	현재 로캘의 통화 및 숫자 형식 정보 얻기

다음 표는 libc의 문자 분류 함수 API를 설명합니다.

표 2-7 libc의 문자 분류 및 자역

라이브러리 루틴	설명
isalpha()	문자가 알파벳인가?
isupper()	문자가 대문자인가?
islower()	문자가 소문자인가?
isdigit()	문자가 숫자인가?
isxdigit()	문자가 16진수인가?
isalnum()	문자가 알파벳인가 숫자인가?
isspace()	문자가 공백인가?
ispunct()	문자가 구두점인가?
isprint()	문자가 인쇄 가능한가?
iscntrl()	문자가 제어 문자인가?
isascii()	문자가 ASCII 문자인가?
isgraph()	문자가 가시 문자인가?
isphonogram()	넓은 문자가 표음 문자인가?
isideogram()	넓은 문자가 표의 문자인가?
isenglish()	영어 알파벳의 넓은 문자가 추가 코드 세트에 왔는가?
isnumber()	넓은 문자가 추가 코드 세트의 숫자인가?
isspecial()	특수 넓은 문자가 추가 코드 세트에 있는가?
iswalpha()	넓은 문자가 알파벳인가?
iswupper()	넓은 문자가 대문자인가?
iswlower()	넓은 문자가 소문자인가?
iswdigit()	문자가 숫자인가?
iswxdigit()	넓은 문자가 6진수인가?

표 2-7 libc의 문자 분류 및 지역 (계속)

라이브러리 루틴	설명
iswalnum()	넓은 문자가 알파벳 문자인가 숫자인가?
iswspace()	넓은 문자가 빈 공백인가?
iswpunct()	넓은 문자가 구두점인가?
iswprint()	넓은 문자가 인쇄 가능 문자인가?
iswgraph()	넓은 문자가 가시 문자인가?
iswcntrl()	넓은 문자가 제어 문자인가?
iswascii()	넓은 문자가 ASCII 문자인가?
toupper()	소문자를 대문자로 변환합니다.
tolower()	대문자를 소문자로 변환합니다.
towupper()	소문자 넓은 문자를 대문자로 변환합니다.
towlower()	대문자 넓은 문자를 소문자로 변환합니다.
towctrans()	넓은 문자 매핑

다음 표는 libc의 문자 조합 함수 API를 설명합니다.

표 2-8 libc의 문자 조합

라이브러리 루틴	설명
strcoll()	문자열 조합
strxfrm()	비교를 위해 문자열 변환
wscoll()	넓은 문자열 조합
wcsxfrm()	비교를 위해 넓은 문자열 변환

다음 표는 libc의 통화 처리 함수 API를 설명합니다.

표 2-9 libc의 통화 형식

라이브러리 루틴	설명
localeconv()	현재 로케일의 통화 형식 정보 얻기
strfmon()	통화 값을 문자열 표현으로 변환

다음 표는 libc의 날짜 및 시간 형식을 설명합니다.

표 2-10 libc의 날짜 및 시간 형식

라이브러리 루틴	설명
getdate()	사용자 형식 날짜 및 시간을 변환
strftime()	날짜 및 시간을 문자열 표현으로 변환 %u 변환 함수는 X/Open CAE Specification, System Interfaces and Headers, Issue 4, 버전 2를 준수합니다. 이 함수는 10진수 [1,7]로 주일을 표현하고 1이 현재 월요일을 나타냅니다.
strptime()	날짜 및 시간 변환

다음 표는 libc의 멀티바이트 처리 함수 API를 설명합니다.

표 2-11 libc의 멀티바이트 처리

라이브러리 루틴	설명
btowc()	단일바이트를 넓은 문자로 변환
mbrlen()	문자의 바이트 수 얻기(다시 시작 가능)
mbsinit()	변환 객체 상태 결정
mbrtowc()	문자를 넓은 문자 코드로 변환(다시 시작 가능)
mbsrtowcs()	문자열을 넓은 문자열로 변환(다시 시작 가능)
mblen()	문자의 바이트 수 얻기
mbtowc()	문자를 넓은 문자 코드로 변환
mbstowcs()	문자열을 넓은 문자열로 변환

다음 표는 libc의 넓은 문자 및 문자열 처리를 설명합니다.

표 2-12 libc의 넓은 문자 및 문자열 처리

라이브러리 루틴	설명
wcsncat()	넓은 문자를 길이 $n$ 으로 연결
wsdup()	넓은 문자열 중복
wscmp()	넓은 문자열 비교
wcsncmp()	넓은 문자를 길이 $n$ 과 비교
wscpy()	넓은 문자열 복사
wcsncpy()	넓은 문자를 길이 $n$ 으로 복사
wcschr()	넓은 문자열에서 문자 찾기

표 2-12 libc의 넓은 문자 및 문자열 처리 (계속)

라이브러리 루틴	설명
wcsrchr()	오른쪽부터 넓은 문자열에서 문자 찾기
wcslen()	넓은 문자열 길이 얻기
wscol()	넓은 문자열의 표시 너비 반환
wcsspn()	다른 넓은 문자열에 있는 하나의 넓은 문자열 범위 반환
wcscspn()	다른 넓은 문자열에 없는 하나의 넓은 문자열 범위 반환
wcspbrk()	다른 넓은 문자열에 없는 하나의 넓은 문자열 포인터 반환
wcstok()	넓은 문자열을 통해 토큰 이동
wscwcs()	넓은 문자열에서 문자열 찾기
wcstombs()	넓은 문자열을 멀티바이트 문자열로 변환
wctomb()	넓은 문자를 멀티바이트 문자로 변환
wcwidth()	넓은 문자의 열 위치 수 결정
wcswidth()	넓은 문자열의 열 위치 수 결정
wctob()	넓은 문자를 단일바이트로 변환
wcrtomb()	넓은 문자를 문자로 변환(다시 시작 가능)
wcstol()	넓은 문자열을 긴 정수로 변환
wcstoul()	넓은 문자열을 부호없는 긴 정수로 변환
wcstod()	넓은 문자열을 배정도로 변환
wcsrtombs()	넓은 문자열을 문자열로 변환(다시 시작 가능)
wcscat()	넓은 문자열 연결

다음 표는 libc의 형식화된 넓은 문자 입출력을 설명합니다.

표 2-13 libc의 형식화된 넓은 문자 입출력

라이브러리 루틴	설명
wsprintf()	형식에 따라 넓은 문자열 생성
wsscanf()	형식화된 입력 변환
fwprintf()	형식화된 넓은 문자 출력 인쇄
fwscanf()	형식화된 넓은 문자 입력 변환



표 2-13 libc의 형식화된 넓은 문자 입출력 (계속)

라이브러리 루틴	설명
wprintf()	형식화된 넓은 문자 출력 인쇄
wscanf()	형식화된 넓은 문자 입력 변환
swprintf()	형식화된 넓은 문자 출력 인쇄
swscanf()	형식화된 넓은 문자 입력 변환
vwprintf()	stdarg 인자 목록의 넓은 문자 형식화된 출력
vswprintf()	stdarg 인자 목록의 넓은 문자 형식화된 출력

이 표는 libc의 넓은 문자열 함수 API를 설명합니다.

표 2-14 넓은 문자열libc

라이브러리 루틴	설명
wscasecmp()	넓은 문자열을 비교하고 대소문자 차이를 무시
wncasecmp()	코드 문자열 작업 프로세스
wcsstr()	넓은 문자열 부속 문자열 찾기
wmemchr()	메모리의 넓은 문자 찾기
wmemcmp()	메모리의 넓은 문자 비교
wmemcpy()	메모리의 넓은 문자 복사
wmemmove()	겹친 영역이 있는 메모리의 넓은 문자 복사
wmemset()	메모리의 넓은 문자 설정

다음 표는 libc의 넓은 문자 입출력을 설명합니다.

표 2-15 libc의 넓은 문자 입출력

라이브러리 루틴	설명
fgetwc()	스트림에서 멀티바이트 문자를 얻어 넓은 문자로 변환
getwchar()	stdin에서 멀티바이트 문자를 얻어 넓은 문자로 변환
fgetws()	스트림에서 멀티바이트 문자열을 얻어 넓은 문자로 변환
getws()	stdin에서 멀티바이트 문자열을 얻어 넓은 문자로 변환
fputwc()	넓은 문자열을 멀티바이트 문자로 변환하여 스트림에 넣기
fwide()	스트림 방향 설정

표 2-15 libc의 넓은 문자 입출력 (계속)

라이브러리 루틴	설명
putwchar()	넓은 문자열을 멀티바이트 문자로 변환하여 stdin에 넣기
fputws()	넓은 문자를 멀티바이트 문자열로 변환하여 스트림에 넣기
putws()	넓은 문자를 멀티바이트 문자열로 변환하여 stdin에 넣기
ungetwc()	넓은 문자를 입력 스트림으로 푸시백

## genmsg 유틸리티

새 genmsg 유틸리티는 catgets() 계열의 함수들이 사용할 수 있는, 국제화된 소스 메시지 카탈로그를 만드는데 사용됩니다. 유틸리티는 catgets의 함수에 대한 호출을 위해 소스 프로그램 파일을 조사하고 찾은 정보에서 소스 메시지 카탈로그를 구축합니다. 예를 들어, 다음과 같습니다.

```
% cat example.c
...
/* NOTE: %s is a file name */
printf(catgets(catd, 5, 1, "%s cannot be opened."));
/* NOTE: "Read" is a past participle, not a present
tense verb */
printf(catgets(catd, 5, 1, "Read"));
...
% genmsg -c NOTE example.c
The following file(s) have been created.
new msg file = "example.c.msg"
% cat example.c.msg
$quote "
$set 5
1          "%s cannot be opened"
/* NOTE: %s is a file name */
2          "Read"
/* NOTE: "Read" is a past participle, not a present
tense verb */
```

위 예제에서, genmsg를 소스 파일 example.c에 실행하여, example.c.msg라는 소스 메시지 카탈로그를 생성해 냈습니다. 인자 NOTE를 옵션 -c와 함께 사용하여 genmsg가 카탈로그 안에 주석을 포함하도록 합니다. 소스 프로그램의 주석이 지정된 문자열을 포함하면 주석은 메시지 카탈로그에서 catgets에 대한 호출에서 추출된 다음 문자열 후에 나타납니다.

genmsg를 사용하여 메시지 세트에서 메시지를 자동으로 번호 매길 수 있습니다.

자세한 내용은 genmsg(1) 설명서 페이지를 참조하십시오.

형식화된 메시지 카탈로그 파일을 생성하려면 `gencat (1)` 유틸리티를 사용하십시오.

이식 가능 메시지 파일(.po 파일)에 대한 메시지 추출 유틸리티와 .po 파일에서 메시지 객체 파일(.mo 파일)을 생성하는 방법에 대한 자세한 내용은 각각 `xgettext(1)`과 `msgfmt(1)` 설명서 페이지를 참조하십시오.

---

## 사용자 정의 및 사용자 확장 가능 코드 변환

Solaris 사용자는 `geniconvtbl` 유틸리티를 사용하여 사용자 정의 코드 세트 변환기를 작성할 수 있습니다.

이 유틸리티는 `iconv(1)`과 `iconv(3C)`와 같은 표준 시스템 유틸리티와 인터페이스로 사용자 정의 및 사용자 정의 가능 코드 세트 변환을 할 수 있습니다. 이 기능은 응용프로그램이 호환되지 않는 데이터 유형, 특히 독점 소유 또는 구형 응용프로그램의 데이터를 처리하는 능력을 향상시킵니다. 기존 Solaris 코드 세트 변환에 대한 수정도 지원됩니다.

자세한 내용과 예제는 `geniconvtbl (1)`과 `geniconvtbl(4)` 설명서 페이지에서 찾을 수 있습니다. 유틸리티에 대한 예제 입력 소스 파일은 `/usr/lib/iconv/geniconvtbl/srcs/` 디렉토리의 참조로도 사용 가능합니다.

사용자 정의 코드 변환이 준비되고 `geniconvtbl(1)` 설명서 페이지에서 지정된대로 배치되면 사용자는 32비트 및 64비트 Solaris 운영 환경의 `iconv(1)` 유틸리티 및 `iconv(3C)` 함수의 코드 변환을 사용할 수 있습니다.



# Solaris 9 환경에서 지역화

---

이 절에서는 Solaris 9 환경에 포함되어 있는 몇 가지 지역화 기능에 대해 논의합니다.

- “지역화를 위한 소프트웨어 지원” 53 페이지
- “지원되는 로켈” 55 페이지
- “로켈용 복수 키 구성 시퀀스” 61 페이지
- “Solaris 9 제품에서의 키보드 지원” 62 페이지

---

## 지역화를 위한 소프트웨어 지원

이 절에는 Solaris 9 로켈 패키지, Solaris 9 CD-ROM의 지역화 콘텐츠, 인터페이스의 지역화 기능 및 스크립트 활성화 등에 관한 정보가 포함되어 있습니다.

## Solaris 9 로켈 패키지 요약

모든 Solaris 9 로켈 패키지는 두 개의 범주로 구분됩니다.

첫 번째 범주는 부분 로켈로, 로켈 사용을 가능케 해줍니다. 시스템에 설치되어 있는 부분 로켈을 통해 사용자는 대상 로켈로 텍스트를 입력, 표시, 인쇄하고 응용프로그램을 실행할 수 있습니다. 하지만 Solaris의 OS/GUI 메시지는 영어로 표현됩니다. 모든 부분 로켈 패키지는 Solaris Software CD에 들어 있습니다. 일본어 및 아시아어 부분 로켈은 해당 언어에 따라 패키지화되며 다른 부분 로켈은 지역에 따라 패키지화됩니다.

두 번째 범주는 전체 로켈 패키지용입니다. 이 패키지에는 소프트웨어 메시지, 온라인 도움말 파일, 옵션 클립 및 언어별 기능이 번역되어 포함되어 있습니다. 전체 로켈 패키지는 여러 언어에 전체 언어 기능 모음을 제공합니다. 다음 언어를 바탕으로 한 모든 로켈은 전체 로켈입니다:

- 독일어

- 프랑스어
- 스페인어
- 스웨덴어
- 이탈리아어
- 일본어
- 한국어
- 간체 한자
- 정체 한자

전체 로케일 패키지는 언어별로 패키지가되며 Language CD에서 사용할 수 있습니다.

---

주 - 전체 로케일이 작동되려면 부분 로케일 패키지(로케일 허가자)를 설치해야 합니다.

---

Solaris 설치 프로세스에서 지원이 필요한 지역을 선택하라는 메시지가 나타납니다. 설치가 완료된 뒤에 사용 가능한 로케일 지원은 이 단계에서의 선택에 따라 달라집니다. 부분 로케일은 Solaris 9 운영 환경이 포함된 Solaris 소프트웨어 CD-ROM에서 설치하며 전체 로케일은 Language CD에서 설치합니다. 전체 로케일 지원이 필요치 않으면 설치 프로세스 중에 Language CD-ROM에서 설치를 건너뛸 수 있습니다. 영어 로케일은 기본 설치된다는 점에 주의하십시오.

## 본 릴리스에서 추가 로케일

이 릴리스용의 새 부분 로케일로 러시아어와 폴란드어용 UTF-8 로케일, 카탈로니아어용 로케일 두 개, 새 태국어 로케일, 새 인도어 로케일, 새 정체 한자 로케일 두 개 및 새 간체 한자 로케일이 추가되었습니다. 로케일 이름은 다음과 같습니다:

- ar\_EG.UTF-8
- ca\_ES.ISO8859-1
- ca\_ES.ISO8859-15
- fi\_FI.UTF-8
- fr\_BE.UTF-8
- pl\_PL.UTF-8
- pt\_BR.UTF-8
- ru\_RU.UTF-8
- tr\_TR.UTF-8
- th\_TH.UTF-8
- hi\_IN.UTF-8
- zh\_HK.BIG5HK

이것은 정체 한자(홍콩) Big5-HKSCS 로케일입니다. 이것은 Language CD에서 정체 한자 메시지 패키지를 설치하면 전체 로케일입니다.

- zh\_HK.UTF-8  
정체 한자(홍콩) Big5—8 로캘입니다. 이것은 Language CD에서 정체 한자 메시지 패키지를 설치하면 전체 로캘입니다.
- zh\_CN.GB18030  
간체 한자 GB18030 로캘입니다. 이것은 Language CD에서 간체 한자 메시지 패키지를 설치하면 전체 로캘입니다.

## 지원되는 로캘

다음 표는 Solaris 9 환경에서 지원되는 모든 로캘을 보여줍니다. 로캘 이름은 국제 이름 표준에 맞추어 업데이트되었습니다.

표 3-1 아시아

로캘	사용자 인터페이스	지역	코드 세트	언어 지원
hi_IN.UTF-8	영어	인도어	UTF-8 <sup>1</sup>	힌두어(UTF-8) 유니 코드 3.1
ja	일본어	일본	eucJP <sup>2</sup>	일본어(EUC) JIS X 0201-1976 JIS X 0208-1990 JIS X 0212-1990
ja_JP.eucJP	일본어	일본	eucJP	일본어(EUC) JIS X 0201-1976 JIS X 0208-1990 JIS X 0212-1990
ja_JP.PCK	일본어	일본	PCK <sup>3</sup>	일본어 (PC kanji) JIS X 0201-1976 JIS X 0208-1990
ja_JP.UTF-8	일본어	일본	UTF-8	일본어 (UTF-8) 유니 코드 3.1
ko_KR.EUC	한국어	한국	1001	한국어 (EUC) KS X 1001
ko_KR.UTF-8	한국어	한국	UTF-8	한국어 (UTF-8) 유니 코드 3.1

UTF-8은 ISO/IEC 10646-1:2000 및 유니 코드 3.1에서 정의된 UTF-8입니다.

eucJP는 일본어 EUC 코드 세트를 의미합니다. ja\_JP.eucJP 로캘 사양은 UI\_OSF Japanese Environment Implementation Agreement 버전 1.1을 준수하며 ja 로캘은 이전 Solaris 릴리스의 전통적 사양을 준수합니다.

PCK는 Shift\_JIS (SJIS)로도 알려져 있습니다.

표 3-1 아시아 (계속)

로케일	사용자 인터페이스	지역	코드 세트	언어 지원
th_TH.UTF-8	영어	태국	UTF-8	태국어 (UTF-8) 유니 코드 3.1
th_TH.TIS620	영어	태국	TIS620.2533	태국어 TIS620.2533
zh_CN.EUC	간체 한자	PRC	gb2312 <sup>4</sup>	간체 한자 (EUC) GB2312-1980
zh_CN.GBK	간체 한자	PRC	GBK <sup>5</sup>	간체 한자 (GBK)
zh_CN.GB18030	간체 한자	PRC	GB18030-2000	간체 중국어(GB18030-2000) GB18030-2000
zh_CN.UTF-8	간체 한자	PRC	UTF-8	간체 한자(UTF-8) 유니 코드 3.1
zh_HK.BIG5HK	정체 한자	홍콩	Big5+HKSCS	정체 한자(BIG5+HKSCS)
zh_HK.UTF-8	정체 한자	홍콩	UTF-8	정체 한자(UTF-8) 유니 코드 3.1
zh_TW.EUC	정체 한자	대만	cns11643	정체 한자 (EUC) CNS 11643-1992
zh_TW.BIG5	정체 한자	대만	BIG5	정체 한자 (BIG5)
zh_TW.UTF-8	정체 한자	대만	UTF-8	정체 한자(UTF-8) 유니 코드 3.1

gb2312는 GB 1988-80 및 GB 2312-80이 포함된 간체 한자 EUC 코드 세트를 의미합니다.

GBK는 GB 확장을 의미합니다. 여기에는 모든 GB 2312-80 문자와 ISO/IEC 10646-1의 모든 Unified Han 문자, 그리고 일본어 히라가나 및 가타카나 문자가 포함되어 있습니다. 또한 중국어, 일본어 및 한국어 문자 세트와 ISO/IEC 10646-1의 많은 문자들이 포함되어 있습니다

표 3-2 남양주

로케일	사용자 인터페이스	지역	코드 세트	언어 지원
en_AU.ISO8859-1	영어	호주	ISO8859-1	영어(호주)
en_NZ.ISO8859-1	영어	뉴질랜드	ISO8859-1	영어(뉴질랜드)

표 3-3 중앙 아메리카

로케일	사용자 인터페이스	지역	코드 세트	언어 지원
es_CR.ISO8859-1	스페인어	코스타리카	ISO8859-1	스페인어(코스타리카)
es_GT.ISO8859-1	스페인어	과테말라	ISO8859-1	스페인어(과테말라)
es_NI.ISO8859-1	스페인어	니카라과	ISO8859-1	스페인어(니카라과)
es_PA.ISO8859-1	스페인어	파나마	ISO8859-1	스페인어(파나마)
es_SV.ISO8859-1	스페인어	엘살바도르	ISO8859-1	스페인어 (엘살바도르)



표 3-4 중부 유럽

로케	사용자 인터페이스	지역	코드 세트	언어 지원
cs_CZ.ISO8859-2	영어	체코 공화국	ISO8859-2	체코어(체코 공화국)
de_AT.ISO8859-1	독일어	오스트리아	ISO8859-1	독일어 (오스트리아)
de_AT.ISO8859-15	독일어	오스트리아	ISO8859-15	독일어 (오스트리아, ISO8859-15 - Euro)
de_CH.ISO8859-1	독일어	스위스	ISO8859-1	독일어 (스위스)
de_DE.UTF-8	독일어	독일	UTF-8	독일어 (독일, 유니 코드 3.1)
de_DE.ISO8859-1	독일어	독일	ISO8859-1	독일어 (독일)
de_DE.ISO8859-15	독일어	독일	ISO8859-15	독일어 (독일, ISO8859-15 - Euro)
fr_CH.ISO8859-1	프랑스어	스위스	ISO8859-1	프랑스어 (스위스)
hu_HU.ISO8859-2	영어	헝가리	ISO8859-2	헝가리어 (헝가리)
pl_PL.ISO8859-2	영어	폴란드	ISO8859-2	폴란드어(폴란드)
pl_PL.UTF-8	영어	폴란드	UTF-8	폴란드어 (폴란드, 유니 코드 3.1)
sk_SK.ISO8859-2	영어	슬로바키아	ISO8859-2	슬로바키아어 (슬로바키아)

표 3-5 동유럽

로케	사용자 인터페이스	지역	코드 세트	언어 지원
bg_BG.ISO8859-5	영어	불가리아	ISO8859-5	불가리아어 (불가리아)
et_EE.ISO8859-15	영어	에스토니아	ISO8859-15	에스토니아어 (에스토니아)
hr_HR.ISO8859-2	영어	크로아티아	ISO8859-2	크로아티아어 (크로아티아)
lt_LT.ISO8859-13	영어	리투아니아	ISO8859-13	리투아니아어 (리투아니아)
lv_LV.ISO8859-13	영어	라트비아	ISO8859-13	라트비아어 (라트비아)
mk_MK.ISO8859-5	영어	마케도니아	ISO8859-5	마케도니아어 (마케도니아)
ro_RO.ISO8859-2	영어	루마니아	ISO8859-2	루마니아어 (루마니아)

표 3-5 동유럽 (계속)

로캘	사용자 인터페이스	지역	코드 세트	언어 지원
ru_RU.KOI8-R	영어	러시아	KOI8-R	러시아어 (러시아, KOI8-R)
ru_RU.ANSI1251	영어	러시아	ansi-1251	러시아어 (러시아, ANSI 1251)
ru_RU.ISO8859-5	영어	러시아	ISO8859-5	러시아어 (러시아)
ru_RU.UTF-8	영어	러시아	UTF-8	러시아어 (러시아, 유니 코드 3.1)
sh_BA.ISO8859-2@bosnia	영어	보스니아	ISO8859-2	보스니아어 (보스니아)
sl_SI.ISO8859-2	영어	슬로베니아	ISO8859-2	슬로베니아어 (슬로베니아)
sq_AL.ISO8859-2	영어	알바니아	ISO8859-2	알바니아어 (알바니아)
sr_YU.ISO8859-5	영어	세르비아	ISO8859-5	세르비아어 (세르비아)
tr_TR.ISO8859-9	영어	터키	ISO8859-9	터키어 (터키)
tr_TR.UTF-8	영어	터키	UTF-8	터키어 (터키, 유니 코드 3.1)

표 3-6 중동

로캘	사용자 인터페이스	지역	코드 세트	언어 지원
He	영어	이스라엘	ISO8859-8	히브리어 (이스라엘)

표 3-7 북 아프리카

로캘	사용자 인터페이스	지역	코드 세트	언어 지원
ar_EG.UTF-8	영어	이집트	UTF-8	아랍어 (이집트)
Ar	영어	이집트	ISO8859-6	아랍어 (이집트)

표 3-8 북아메리카

로캘	사용자 인터페이스	지역	코드 세트	언어 지원
en_CA.ISO8859-1	영어	캐나다	ISO8859-1	영어 (캐나다)
en_US.ISO8859-1	영어	USA	ISO8859-1	영어 (미국)

표 3-8 북아메리카 (계속)

로캘	사용자 인터페이스	지역	코드 세트	언어 지원
en_US.ISO8859-15	영어	USA	ISO8859-15	영어 (미국, ISO8859-15 - Euro)
en_US.UTF-8	영어	USA	UTF-8	영어 (미국, 유니 코드 3.1)
fr_CA.ISO8859-1	프랑스어	캐나다	ISO8859-1	프랑스어 (캐나다)
es_MX.ISO8859-1	스페인어	멕시코	ISO8859-1	스페인어 (멕시코)

표 3-9 북유럽

로캘	사용자 인터페이스	지역	코드 세트	언어 지원
da_DK.ISO8859-1	영어	덴마크	ISO8859-1	덴마크어 (덴마크)
da_DK.ISO8859-15	영어	덴마크	ISO8859-15	덴마크어 (덴마크, ISO8859-15-Euro)
fi_FI.ISO8859-1	영어	핀란드	ISO8859-1	핀란드어, 유니 코드 3.1)
fi_FI.ISO8859-15	영어	핀란드	ISO8859-15	핀란드어 (핀란드, ISO8859-15-Euro)
fi_FI.UTF-8	영어	핀란드	UTF-8	핀란드어 (핀란드)
is_IS.ISO8859-1	영어	아이슬란드	ISO8859-1	아이슬란드어 (아이슬란드)
no_NO.ISO8859-1@bokmal	영어	노르웨이	ISO8859-1	노르웨이어 (노르웨이-북말)
no_NO.ISO8859-1@nyorsk	영어	노르웨이	ISO8859-1	노르웨이어 (노르웨이-니노르스크)
sv_SE.ISO8859-1	스웨덴어	스웨덴	ISO8859-1	스웨덴어 (스웨덴)
sv_SE.ISO8859-15	스웨덴어	스웨덴	ISO8859-15	스웨덴어 (스웨덴, ISO8859-15-Euro)
sv_SE.UTF-8	스웨덴어	스웨덴	UTF-8	스웨덴어 (스웨덴, 유니코드 3.1)

표 3-10 남아메리카

로캘	사용자 인터페이스	지역	코드 세트	언어 지원
es_AR.ISO8859-1	스페인어	아르헨티나	ISO8859-1	스페인어 (아르헨티나)
es_BO.ISO8859-1	스페인어	볼리비아	ISO8859-1	스페인어 (볼리비아)
es_CL.ISO8859-1	스페인어	칠레	ISO8859-1	스페인어 (칠레)
es_CO.ISO8859-1	스페인어	콜롬비아	ISO8859-1	스페인어(콜롬비아)

표 3-10 남아메리카 (계속)

로케	사용자 인터페이스	지역	코드 세트	언어 지원
es_EC.ISO8859-1	스페인어	에쿠아도르	ISO8859-1	스페인어 (에쿠아도르)
es_PE.ISO8859-1	스페인어	페루	ISO8859-1	스페인어 (페루)
es_PY.ISO8859-1	스페인어	파라과이	ISO8859-1	스페인어 (파라과이)
es_UY.ISO8859-1	스페인어	우루과이	ISO8859-1	스페인어 (우루과이)
es_VE.ISO8859-1	스페인어	베네수엘라	ISO8859-1	스페인어 (베네수엘라)
pt_BR.ISO8859-1	영어	브라질	ISO8859-1	포르투갈어 (브라질)
pt_BR.UTF-8	영어	브라질	UTF-8	포르투갈어 (브라질, 유니 코드 3.1)

표 3-11 남부 유럽

로케	사용자 인터페이스	지역	코드 세트	언어 지원
ca_ES.ISO8859-1	영어	스페인	ISO8859-1	카탈로니아어 (스페인)
ca_ES.ISO8859-15	영어	스페인	ISO8859-15	카탈로니아어 (스페인, ISO8859-15 - Euro)
el_GR.ISO8859-7	영어	그리스	ISO8859-7	그리스어 (그리스)
es_ES.ISO8859-1	스페인어	스페인	ISO8859-1	스페인어 (스페인)
es_ES.ISO8859-15	스페인어	스페인	ISO8859-15	스페인어 (스페인, ISO8859-15 - Euro)
es_ES.UTF-8	스페인어	스페인	UTF-8	스페인어 (스페인, 유니 코드 3.1)
it_IT.ISO8859-1	이탈리아어	이탈리아	ISO8859-1	이탈리아어 (이탈리아)
it_IT.ISO8859-15	이탈리아어	이탈리아	ISO8859-15	이탈리아어 (이탈리아, ISO8859-15 - Euro)
it_IT.UTF-8	이탈리아어	이탈리아	UTF-8	이탈리아어 (이탈리아, 유니 코드 3.1)
pt_PT.ISO8859-1	영어	포르투갈	ISO8859-1	포르투갈어 (포르투갈)
pt_PT.ISO8859-15	영어	포르투갈	ISO8859-15	포르투갈어 (포르투갈, ISO8859-15 - Euro)

표 3-12 서유럽

로케	사용자 인터페이스	지역	코드 세트	언어 지원
en_GB.ISO8859-1	영어	영국	ISO8859-1	영어 (영국)

표 3-12 서유럽 (계속)

로케일	사용자 인터페이스	지역	코드 세트	언어 지원
en_IE.ISO8859-1	영어	아일랜드	ISO8859-1	영어 (아일랜드)
fr_BE.ISO8859-1	프랑스어	벨기에-왈론	ISO8859-1	프랑스어(벨기에-왈론, 유니 코드 3.1)
fr_BE.UTF-8	프랑스어	벨기에-왈론	UTF-8	프랑스어(벨기에-왈론, 유니 코드 3.1)
fr_FR.ISO8859-1	프랑스어	프랑스	ISO8859-1	프랑스어 (프랑스)
fr_FR.UTF-8	프랑스어	프랑스	UTF-8	프랑스어 (프랑스, 유니 코드 3.1)
nl_BE.ISO8859-1	영어	벨기에-플란더즈	ISO8859-1	네덜란드어 (벨기에-플란더즈)
nl_NL.ISO8859-1	영어	네덜란드	ISO8859-1	네덜란드어 (네덜란드)

## 로케일용 복수 키 구성 시퀀스

많은 Solaris 로케일들, 특히 유럽 및 유니 코드 로케일들은 Compose 키 시퀀스로 알려져 있는 “죽은 키 시퀀스”를 사용한 다양한 문자 입력을 허용합니다.

Compose 키 시퀀스 입력은 구분 기호나 키보드 키 캡에 나타나 있지 않은 기타 문자들을 입력하는데 사용합니다.

다음 표는 Compose 키 시퀀스의 몇 가지 예를 보여줍니다. Compose 키 시퀀스에 관한 자세한 정보는 “영어/유럽어 입력 모드” 97 페이지를 참조하십시오.

표 3-13 Compose 키로 만든 구분 기호

기호	Compose 키 조합	예제
분음 부호	”	Compose A ” → 분음 부호가 있는 A
카론	V	Compose Z v → 카론이 있는 Z
단음 기호	u	Compose G u → 단음 기호가 있는 G
오고넥	a	Compose A a → 오고넥이 있는 A
세딜라	,	Compose K , → 세딜라가 있는 K
등록 기호	R O	Compose R O → 등록 기호
전치 느낌표	!!	Compose !! → 전치 느낌표

주 - 현재 로캘의 코드 세트에 해당 문자가 없으면 compose 시퀀스를 사용할 수 없습니다. 예를 들어, ISO8859-1에는 caron이 있는 z가 없기 때문에 en\_US.ISO8859-1 로캘에서는 caron이 있는 z를 입력할 수 없습니다.

## Solaris 9 제품에서의 키보드 지원

Solaris는 특정 지역용으로 만들어진 키 레이아웃이 다른 다양한 키보드를 인식 및 지원하며 레이아웃은 Sun SPARC 및 Intel 아키텍처 (IA) 플랫폼을 모두 지원합니다. Solaris 9은 다음 표에 등록된 지역 키보드를 지원합니다:

표 3-14 지역 키보드 지원

지역	국가	Sun 키보드 (유형 4/5/5c)	Sun 키보드 (유형 6)	PC 키보드
아시아	일본	X	X	X
	한국	X	X	X
	대만	X	X	X
유럽	벨기에	X	X	X
	체코 공화국	X		X
	덴마크	X	X	X
	핀란드		X	
	프랑스	X	X	X
	독일	X	X	X
	영국	X	X	X
	그리스	X		X
	헝가리	X		X
	이탈리아	X	X	X
	라트비아	X		X
	리투아니아	X		X
	네덜란드	X	X	X
	노르웨이	X	X	X

표 3-14 지역 키보드 지원 (계속)

지역	국가	Sun 키보드 (유형 4/5/5c)	Sun 키보드 (유형 6)	PC 키보드
	폴란드	X		X
	포르투갈	X	X	X
	러시아	X	X	X
	스페인	X	X	X
	스웨덴	X	X	X
	스위스(프랑스어)	X	X	X
	스위스(독일어)	X	X	X
	터키	X	X	X
아메리카	캐나다(프랑스어)	X	X	X
	남미 (스페인어)	X		
	미국	X	X	X
중동	아랍어	X	X	

중국과 같이 국제 표준을 따르는 키보드 레이아웃을 갖고 있는 지역에서는 미국용으로 제공되는 키보드 레이아웃 지원을 이용해 로캘 문자를 입력합니다. 기본 키보드 매핑은 동일합니다. 일본, 터키 및 스위스와 같은 일부 국가들은 여러 언어를 사용하거나 복수 키보드 레이아웃이 존재하므로 여러 개의 키보드를 갖고 있습니다.

Sun 유형 4, 5 및 5c 키보드는 Mini DIN 8-핀 연결을 통해 Sun I/O 인터페이스를 사용합니다. Sun 유형 6 키보드의 인터페이스 버전은 두 가지입니다:

- Mini DIN 8-핀 연결을 통한 Sun I/O
- USB

Sun 키보드 유형은 각각의 Sun 키보드 뒷면에 인쇄되어 있습니다.

PC 키보드는 PS/2나 USB와 같은 다양한 인터페이스를 사용합니다.

## SPARC 시스템에서 키보드 변경하기

Sun 유형 4, 5 및 5c 키보드 대부분의 DIP 스위치 설정을 사용해 Solaris 제품의 키보드 레이아웃을 변경할 수 있습니다. DIP 스위치 설정을 위해 사용할 수 있는 키보드 유형, 이름 및 관련 레이아웃 id는

`/usr/openwin/share/etc/keytables/keytable.map` 파일에 있습니다.

주 - 키보드 뒷면에 DIP 스위치가 없기 때문에 유형 6 키보드의 레이아웃은 변경할 수 없습니다. 일부 유형 5 및 5c 키보드, 예를 들어 미국, 미국/UNIX 및 일본어 키보드에는 DIP 스위치 대신 접퍼가 있습니다. 키보드 변경을 위한 SPARC 및 IA 플랫폼 모두를 위한 유틸리티나 도구는 Solaris 9 운영 환경에 번들로 제공되지 않습니다( xmodmap(1)과 같은 표준 UNIX 도구는 제외).

다음은 유형 4, 5 및 5c 키보드용 레이아웃 id 값 테이블입니다. (1 = 스위치 켜, 0 = 스위치 끄)

표 3-15 유형 4, 5 및 5c 키보드용 레이아웃

DIP 스위치	키보드 (키 테이블 파일)	바이너리로 설정
0	미국 (US4 .kt)	000000
1	미국 (US4 .kt)	000001
2	벨기에(FranceBelg4 .kt )	000010
3	캐나다 (Canada4 .kt )	000011
4	덴마크 (Denmark4 .kt )	000010
5	독일 (Germany4 .kt )	000101
6	이탈리아 (Italy4 .kt )	000110
7	네덜란드 ( Netherland4 .kt)	000111
8	노르웨이 (Norway4 .kt )	000010
9	포르투갈 (Portugal4 .kt )	001001
10 (0x0a)	남미/스페인어 (SpainLatAm4 .kt)	001010
11 (0x0b)	스웨덴 ( SwedenFin4 .kt)	001011
12 (0x0c)	스위스/프랑스어 (Switzer_Fr4 .kt)	000110
13 (0x0d)	스위스/독일어 (Switzer_Ge4 .kt)	001011
14 (0x0e)	영국 ( UK4 .kt)	001110
16 (0x10)	한국 (Korea4 .kt )	000010
17 (0x11)	대만 ( Taiwan4 .kt)	001001
33 (0x21)	미국 (US5 .kt)	100001
34 (0x22)	미국/UNIX (US_UNIX5 .kt)	100010
35 (0x23)	프랑스 (France5 .kt)	100011



표 3-15 유형 4, 5 및 5c 키보드용 레이아웃 (계속)

DIP 스위치	키보드 (키 테이블 파일)	바이너리로 설정
36 (0x24)	덴마크 (Denmark5.kt)	100010
37 (0x25)	독일 (Germany5.kt)	100101
38 (0x26)	이탈리아 (Italy5.kt)	100110
39 (0x27)	네덜란드 (Netherland5.kt)	100111
40 (0x28)	노르웨이 (Norway5.kt)	101000
41 (0x29)	포르투갈 (Portugal5.kt)	100101
42 (0x2a)	스페인 (Spain5.kt)	101010
43 (0x2b)	스웨덴 (Sweden5.kt)	101011
44 (0x2c)	스위스/프랑스어 (Switzer_Fr5.kt)	101101
45 (0x2d)	스위스/독일어 (Switzer_Ge5.kt)	101110
46 (0x2e)	영국 (UK5.kt)	101111
47 (0x2f)	한국 (Korea5.kt)	101111
48 (0x30)	대만 (Taiwan5.kt)	110000
49 (0x31)	일본 (Japan5.kt)	110001
50 (0x32), 참조: 63 (0x3f)	캐나다/프랑스어 (Canada_Fr5.kt)	110010
51 0(x33)	헝가리 (Hungary5.kt)	110011
52 (0x34)	폴란드 (Poland5.kt)	110100
53 (0x35)	체코 (Czech5.kt)	110101
54 (0x36)	러시아 (Russia5.kt)	110110
55 (0x37)	라트비아 (Latvia5.kt)	110111
56 (0x38) 참조: 62 (0x3e)	터키-Q5 (TurkeyQ5.kt)	000111
57 (0x39)	그리스 (Greece5.kt)	111001
58 (0x3a)	아랍어 (Arabic5.kt)	110111
59 (0x3b)	리투아니아 (Lithuania5.kt)	111010
60 (0x3c)	벨기에 (Belgian5.kt)	111100
62 (0x3e)	터키-F5 (TurkeyF5.kt)	111110
63 (0x3f)	캐나다/프랑스어 (Canada_Fr5_TBITS5.kt)	111111

4가 포함되어 있는 키테이블 파일명은 유형 4 키보드용입니다. 5가 포함되어 있는 키테이블 파일명은 유형 5 키보드용입니다.

키보드 레이아웃을 다른 레이아웃(예: 체코)으로 변경하려면 다음 단계를 따르십시오:

1. 테이블이나 `/usr/openwin/share/etc/keytables/keytable.mp` 파일에서 올바른 DIP 스위치 id(또는 레이아웃 id)를 찾습니다. `keytable.mp` 파일의 레이아웃 id 값은 십진수 값입니다.  
체코의 경우 레이아웃 id는 십진수로 53입니다(16진수로는 0x35).
2. 레이아웃 id를 이진수로 변환하거나 위 테이블의 열에 있는 적당한 "이진수 설정" 값을 사용하십시오. 기수 변환을 위해서는 `dtcalc(1)`과 같은 계산 유틸리티를 사용할 수 있습니다.  
체코 키보드용 올바른 이진 값은 110101입니다.
3. 슈퍼유저가 되십시오. 시스템을 종료한 뒤 전원을 끄십시오.
4. 단계 2의 이진값을 사용해 키보드 뒷면의 DIP 스위치 설정을 변경하십시오.  
첫 번째 DIP 스위치는 왼쪽에 있습니다. "1"의 경우에는 스위치를 up으로 "0"의 경우에는 down으로 옮기십시오.  
체코 키보드 이진 값 110101은 다음에 해당됩니다: 위쪽, 위쪽, 아래쪽, 위쪽, 아래쪽, 위쪽.
5. 시스템 전원을 켜 뒤 부팅시켜 사용하십시오.

---

주 - 유형 4 키보드와 달리 유형 5 및 5c 키보드에는 DIP 스위치가 다섯 개 뿐입니다. 유형 5 및 5c 키보드에서는 첫 번째 이진수는 무시하십시오. 체코 유형 5c 키보드의 경우 올바른 DIP 스위치 설정은 "위쪽, 아래쪽, 위쪽, 아래쪽, 위쪽"이며 10101에서 마지막 다섯 자리만 사용합니다.

---

## Intel 시스템에서 키보드 변경하기

Intel 아키텍처 시스템에서 키보드는 설치 중에 `kdmconfig(1M)` 부분에서 선택합니다. 설치 뒤에 변경하려면 GUI 데스크탑 환경에서 명령줄 모드로 빠져나오십시오. 슈퍼유저로서 `kdmconfig`를 입력해 프로그램을 실행하십시오. 원하는 키보드 레이아웃을 얻으려면 지침에 따르십시오.

### 키보드 레이아웃 도표

다음 그림은 아랍어 키보드를 보여줍니다.

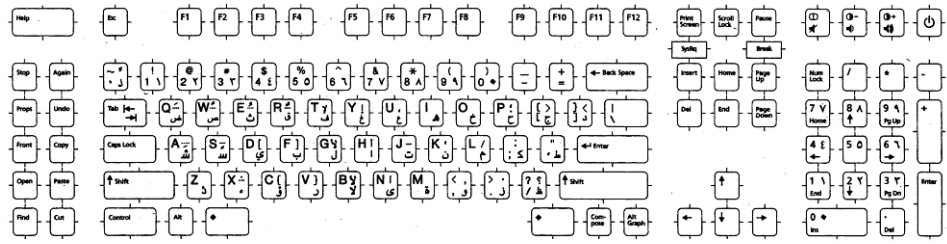


그림 3-1 아랍어 키보드

다음 그림은 벨기에 키보드를 보여줍니다.

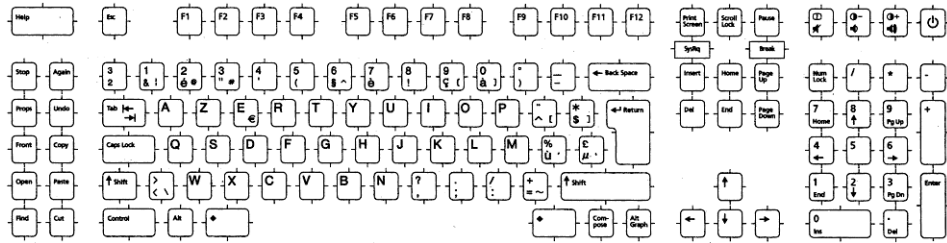


그림 3-2 벨기에 키보드

다음 그림은 키릴어 키보드를 보여줍니다.

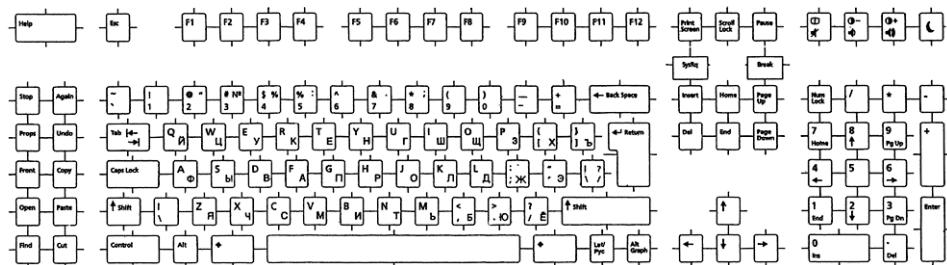


그림 3-3 키릴어(러시아어) 키보드

다음 그림은 덴마크어 키보드를 보여줍니다.

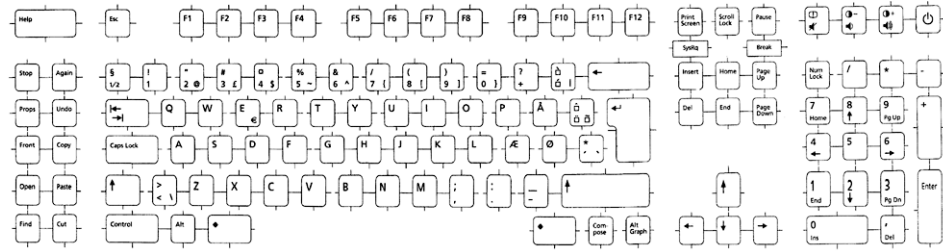


그림 3-4 덴마크어 키보드

다음 그림은 핀란드어 키보드를 보여줍니다.

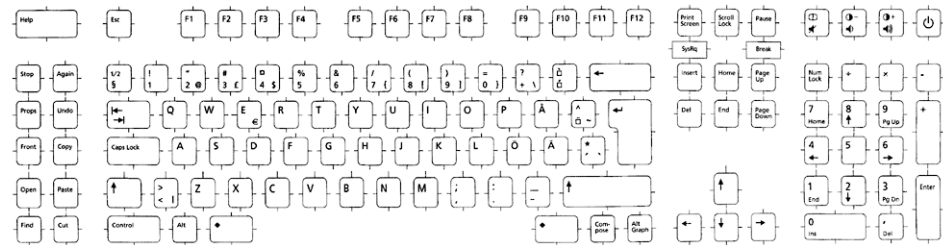


그림 3-5 핀란드어 키보드

다음 그림은 프랑스어 키보드를 보여줍니다.

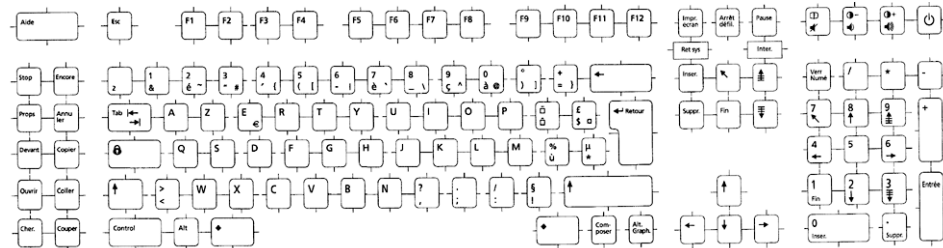


그림 3-6 프랑스어 키보드

다음 그림은 독일어 키보드를 보여줍니다.

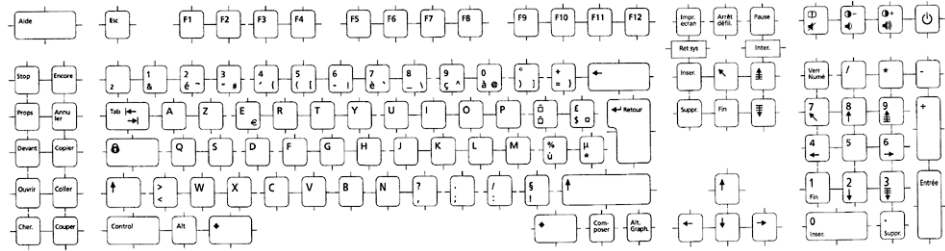


그림 3-7 독일어 키보드

다음 그림은 이탈리아어 키보드를 보여줍니다.

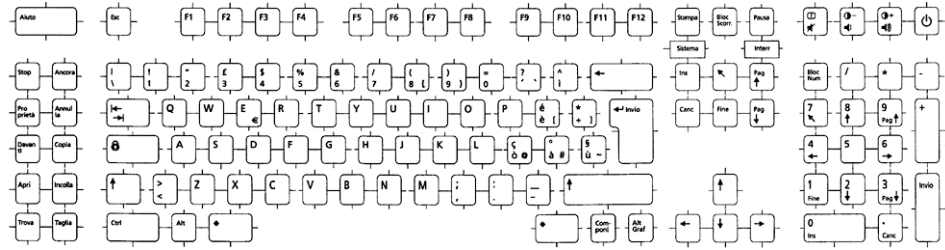


그림 3-8 이탈리아어 키보드

다음 그림은 일본어 키보드를 보여줍니다.

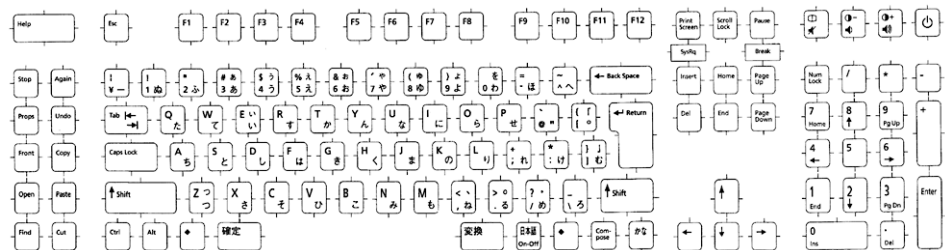


그림 3-9 일본어 키보드

다음은 한국어 키보드를 보여줍니다.

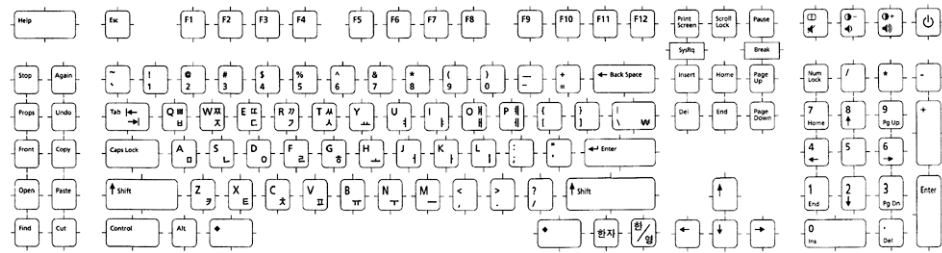


그림 3-10 한국어 키보드

다음은 네덜란드(네덜란드어) 키보드를 보여줍니다,

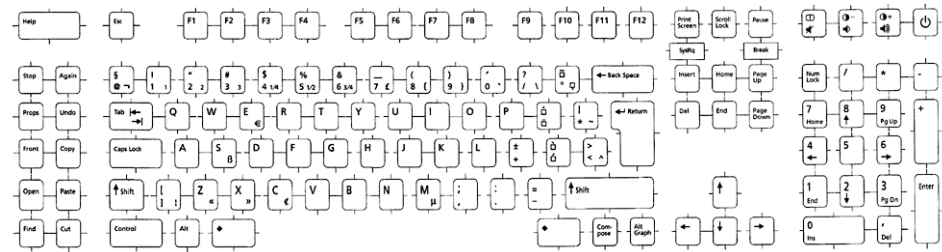


그림 3-11 네덜란드 (네덜란드어) 키보드

다음 그림은 노르웨이어 키보드를 보여줍니다.

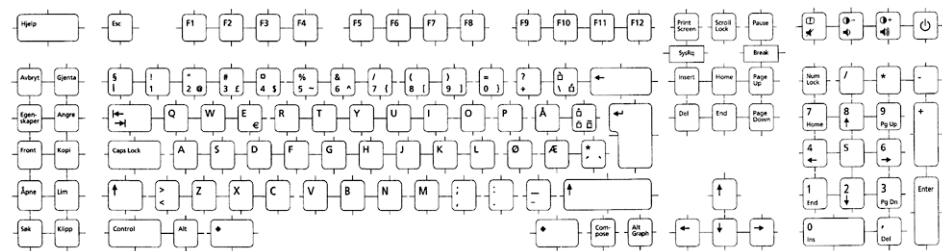


그림 3-12 노르웨이어 키보드

다음 그림은 포르투갈어 키보드를 보여줍니다.

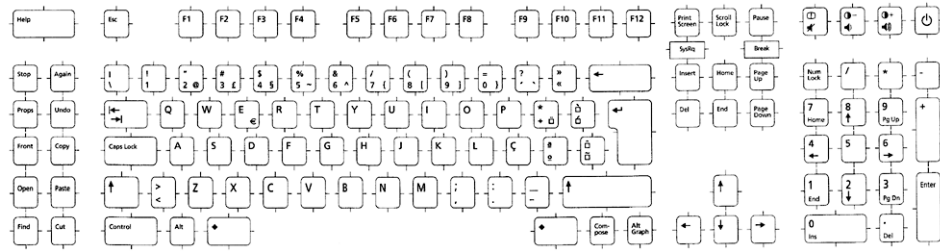


그림 3-13 포르투갈어 키보드

다음 그림은 스페인어 키보드를 보여줍니다.

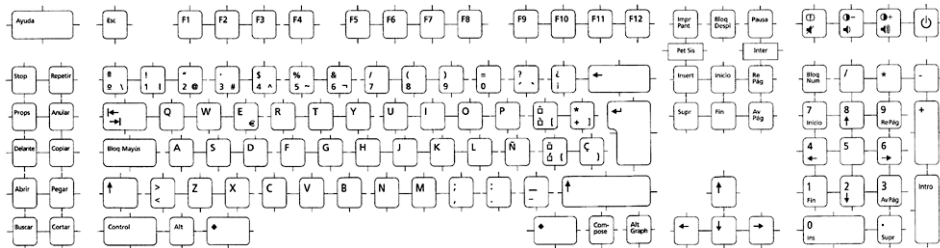


그림 3-14 스페인어 키보드

다음 그림은 스웨덴어 키보드를 보여줍니다.

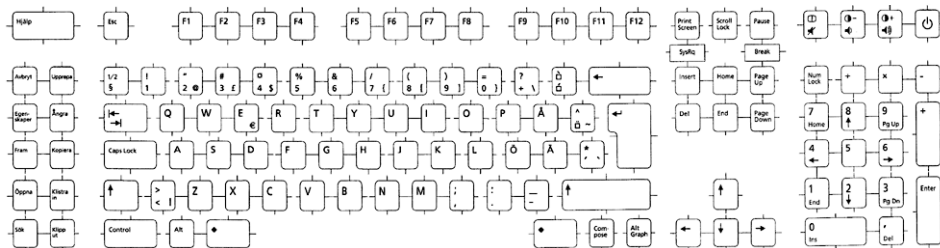


그림 3-15 스웨덴어 키보드

다음 그림은 스위스(프랑스어) 키보드를 보여줍니다.

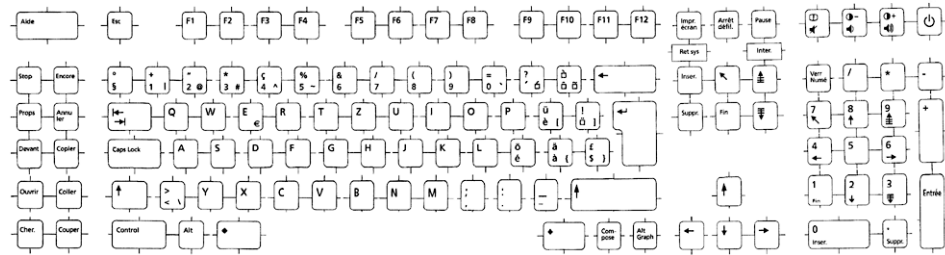


그림 3-16 스위스 (프랑스어) 키보드

다음 그림은 스위스(독일어) 키보드를 보여줍니다.

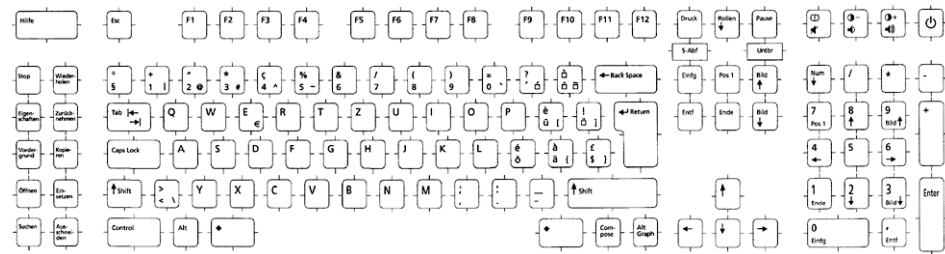


그림 3-17 스위스 (독일어) 키보드

다음 그림은 정체 한자 키보드를 보여줍니다.

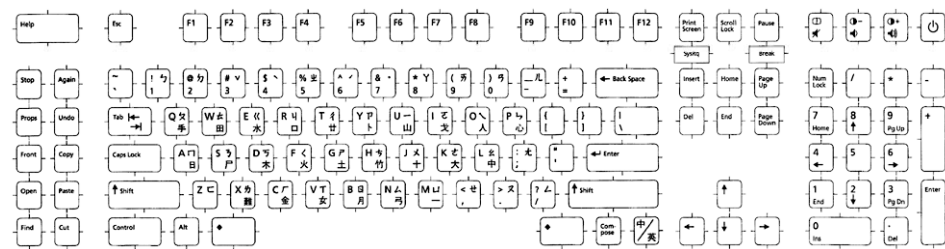


그림 3-18 정체 한자 키보드

다음 그림은 터키어 F 키보드를 보여줍니다.



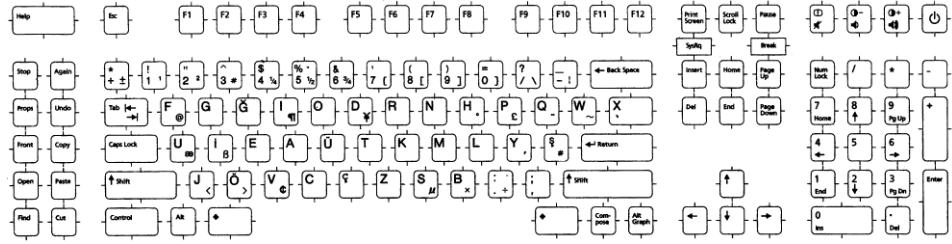


그림 3-19 터키어 F 키보드

다음 그림은 터키어 Q 키보드를 보여줍니다.

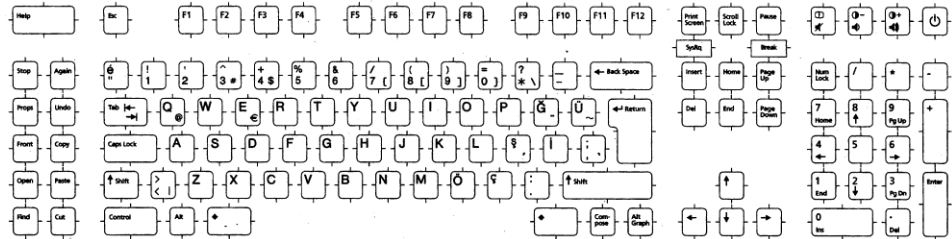


그림 3-20 터키어 Q 키보드

다음 그림은 영국 키보드를 보여줍니다.

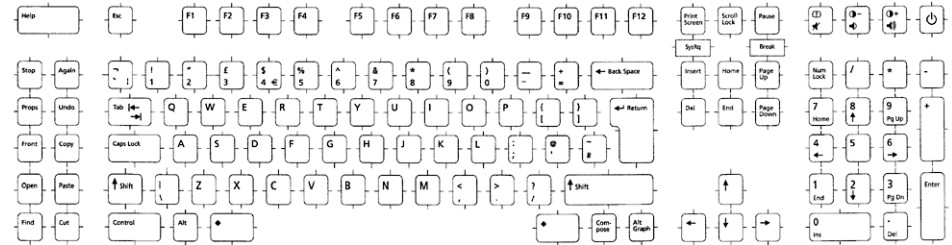


그림 3-21 영국 키보드

다음 그림은 미국 키보드를 보여줍니다.

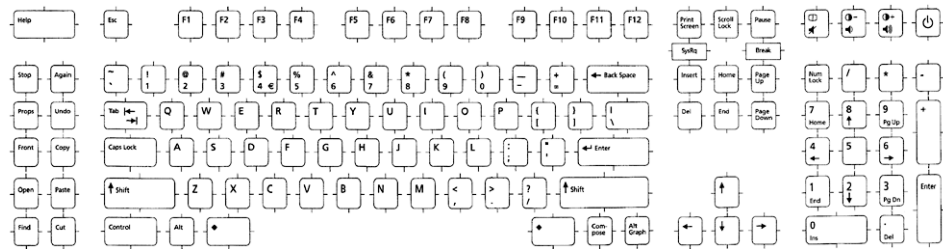


그림 3-22 미국 키보드

다음 그림은 미국/UNIX 키보드를 보여줍니다.

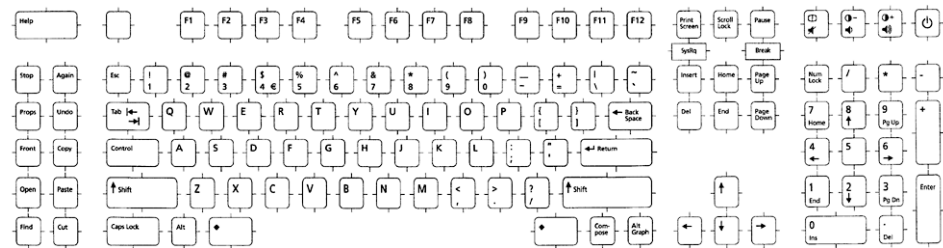


그림 3-23 미국/UNIX 키보드

## 4장

---

# 지원되는 아시아권 로캘

---

다음 절에서는 아시아권에서 지원되는 로캘을 설명합니다:

- “태국어 지역화” 77 페이지
- “간체 한자 지역화” 78 페이지
- “정체 한자 지역화” 81 페이지
- “일본어 지역화” 84 페이지
- “한국어 지역화” 87 페이지

---

## 아시아권에서 지원되는 로캘

다음 표는 아시아권에서 지원되는 로캘을 요약하여 보여줍니다.

표 4-1 아시아권 로캘 요약

언어	로캘 이름	설명	지원되는 문자 세트
한국어	ko	한국어 (EUC)	KS X 1001
	ko.UTF-8	한국어 (UTF-8)	KS X 1005-1
간체 한자	zh_CN.EUC	간체 한자 (EUC)	GB 2312-1980
	zh_CN.GBK	간체 한자 (GBK)	GBK
	zh_CN.GB18030	간체 한자 (GB18030-2000)	GB18030-2000
	zh_CN.UTF-8	간체 한자 (UTF-8)	유니 코드 3.1

---

표 4-1 아시아권 로캘 요약 (계속)

언어	로캘 이름	설명	지원되는 문자 세트
정체 한자	zh_TW.EUC	정체 한자(EUC)	CNS 11643-1992
	zh_TW.UTF-8	정체 한자 (UTF-8)	유니 코드 3.1
	zh_TW.BIG5	정체 한자(BIG5)	BIG5
	zh_HK.BIG5HK	정체 한자 (BIG5+HKSCS)	BIG5+HKSCS
	zh_HK.UTF-8	정체 한자 (UTF-8)	유니 코드 3.1
일본어	ja	일본어(EUC)	JIS <sup>1</sup>
	ja_JP.eucJP	일본어 (EUC)	JIS <sup>2</sup>
	ja_JP.PCK	일본어 (PCK)	유니 코드 3.1
	ja_JP.UTF-8	일본어 (UTF-8)	
태국어	th_TH.TIS620	태국어 (TIS620.2533)	TIS620.2533
	th_TH.UTF-8	태국어 (UTF-8)	유니 코드 3.1
	th_TH.ISO8859-11	태국어 (ISO8859-11)	ISO8859-11
힌디어	hi_IN.UTF-8	힌디어 (UTF-8)	유니 코드 3.1

JIS X 0201-1976, JIS X 0208-1990 및 JIS X 0212-1990

JIS X 0201-1976 및 JIS X 0208-1990

## 간체 및 정체 한자용 입력 방식 보조 창

이 창은 모든 중국 고객들에게 친숙하고 확장성 있는 입력 방식 관리 도구를 제공합니다. 새 입력 방식 보조 창은 다음과 같은 새로운 기능과 유틸리티를 지원합니다:

- 입력 방식 전환
- 중국어 전자/반자 문자 문드 전환
- 중국어/영어 구두점 모드 전환
- 입력 방식 속성 설정
- 입력 방식 선택
- GB2312/GBK/GB18030/CNS11643/Big5/HKSCS/Unicode 문자 세트용 조회 테이블
- 가상 키보드

자세한 정보는 **간체 한자 사용 설명서** 및 **정체 한자 사용 설명서**를 참조하십시오.

이 입력 방식 보조 창은 모든 UTF-8 로캘과 다음 중국어 로캘을 지원합니다:

- zh/zh\_CN.EUC
- zh.GBK/zh\_CN.GBK
- zh.UTF-8/zh\_CN.UTF-8

- zh\_TW/zh\_TW.EUC
- zh\_TW.BIG5
- zh\_TW.UTF-8
- zh\_HK.BIG5HK
- zh\_HK.UTF-8
- zh\_CN.GB18030

두 가지 입력 방식이 지원됩니다:

- 간지와 같은 코드 테이블을 바탕으로 한 메소드
- 판매자가 개발한 메소드(예: NewPinYin 또는 NeiMa)

다음 그림은 보조 창 지원용 인터페이스 모델을 보여줍니다.

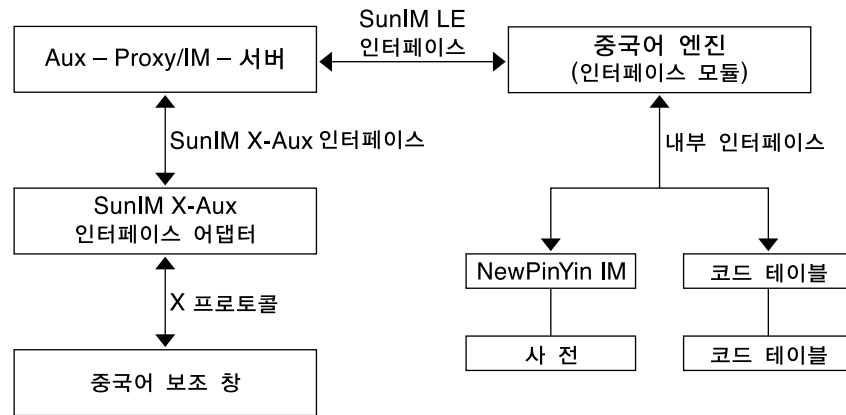


그림 4-1 보조 창 지원용 인터페이스 모델

## 태국어 지역화

태국어 IT 표준에 따르면 태국어 문자 시퀀스 검사 메소드용 입력 레벨은 세 가지가 있습니다:

1. 통과 레벨, 입력 검사 하지 않음.
2. 기본 입력 검사 레벨.
3. 엄격한 입력 검사 레벨.

Solaris 9 릴리스에서 기본 입력 검사 레벨은 아직 통과 레벨입니다. 이것은 시퀀스 검사를 하지 않는다는 것을 의미하며 이전 Solaris 릴리스에서와 동일 레벨입니다. F2 기능을 사용해 세 가지 레벨 간의 전환을 할 수 있습니다:

```
passthrough -> basic -> strict -> passthrough
```

## 태국어 입력 방식 보조 창

태국어 입력 방식 보조 창은 다음과 같은 새로운 기능과 유틸리티를 지원합니다:

- 세 가지 입력 레벨간의 전환(통과/기본/엄격)
- 태국어 가상 키보드

특정 태국어 입력 레벨 및 입력 검사 레벨을 선택하려면 보조 막대의 입력 레벨 버튼을 누르십시오. 태국어 가상 키보드를 표시하려면 키보드 버튼을 누르십시오. 태국어 가상 키보드를 사용하여 태국어 문자를 입력하십시오.

---

## 간체 한자 지역화

Solaris 9 환경의 간체 한자는 네 가지 로캘(zh, zh.GBK, zh\_CN.GB18030 및 zh.UTF-8)을 제공합니다. GB2312-80을 코드화하기 위해서는 zh 로캘에서 EUC 체계가 사용됩니다. zh.GBK 로캘은 GB2312-80의 슈퍼 세트인 GBK 코드 세트를 지원합니다.

새 GB18030-2000 코드 세트는 이제 zh\_CN.GB18030 로캘에서 지원됩니다.

간체 한자는 주로 중국 (PRC)과 싱가포르에서 사용됩니다.

다음 입력 방식은 zh 로캘용으로 지원됩니다:

- 새 QuanPin
- 새 ShuangPin
- QuanPin
- ShuangPin
- GB2312 NeiMa
- 영어-중국어
- 별도의 코드 테이블 입력 방식
- 간체 한자용 입력 방식 보조 창 지원

다음 입력 방식은 zh\_CN.GB18030 로캘용으로 지원됩니다:

- 새 QuanPin
- 새 ShuangPin
- QuanPin
- ShuangPin
- GB18030-2000 NeiMa
- 영어-중국어
- 별도의 코드 테이블 입력 방식
- 간체 한자용 입력 방식 보조 창 지원

다음 입력 방식은 zh.GBK 및 zh.UTF-8 로캘 모두에 지원됩니다:

- 새 QuanPin
- 새 ShuangPin

- QuanPin
- ShuangPin
- GBK NeiMa
- 영어-중국어
- 별도의 코드 테이블 입력 방식
- 간체 한자용 입력 방식 보조 창 지원

중국어 입력 방식용 보조 창은 모든 중국어 로캘에 대해 친숙하고 확장성 있는 입력 방식 사용자 인터페이스를 제공합니다. “간체 및 정체 한자용 입력 방식 보조 창” 76 페이지를 참조하십시오.

중국어 입력 방식용 보조 창에 대한 자세한 정보는 **간체 한자 사용 설명서 및 정체 한자**를 참조하십시오.

다음 표는 zh 로캘용 트루 타입 글꼴을 보여줍니다.

표 4-2 zh\_CN.EUC 로캘용 트루 타입 글꼴

전체 패밀리 이름	하위 패밀리	형식	판매자	코드화
Fangsong	R	트루 타입	Hanyi	GB2312.1980
Hei	R	트루 타입	모노 타입	GB2312.1980
Kai	R	트루 타입	모노 타입	GB2312.1980
Song	R	트루 타입	모노 타입	GB2312.1980

다음 표는 zh 로캘용 비트맵 글꼴을 보여줍니다.

표 4-3 zh\_CN.EUC 로캘용 비트맵 글꼴

전체 패밀리 이름	하위 패밀리	형식	코드화
Song	B	PCF (14,16)	GB2312.1980
Song	R	PCF (12,14,16,20,24)	GB2312.1980

다음 표는 zh\_CN.GBK 로캘용 트루 타입 글꼴을 보여줍니다.

표 4-4 zh\_CN.GBK 로캘용 트루 타입 글꼴

전체 패밀리 이름	하위 패밀리	형식	판매업체	코드화
Fangsong	R	트루 타입	Zhongyi	GBK
Hei	R	트루 타입	Zhongyi	GBK
Kai	R	트루 타입	Zhongyi	GBK
Song	R	트루 타입	Zhongyi	GBK

다음 표는 zh\_CN.GBK 로캘용 비트맵 글꼴을 보여줍니다.

표 4-5 zh\_CN.GBK 로캘용 비트맵 글꼴

전체 패밀리 이름	하위 패밀리	형식	코드화
Song	R	PCF (12,14,16,20,24)	GBK

다음 표는 zh\_CN.GB18030 로캘용 트루 타입 글꼴을 보여줍니다.

표 4-6 zh\_CN.GB18030 로캘용 트루 타입 글꼴

패밀리 이름	하위 패밀리	형식	판매업체	코드화
FangSong	R	트루 타입	FangZheng	GB18030-2000
Song	R	트루 타입	FangZheng	GB18030-2000
Hei	R	트루 타입	FangZheng	GB18030-2000
Kai	R	트루 타입	FangZheng	GB18030-2000

다음 표는 zh\_CN.GB18030 로캘용 비트맵 글꼴을 보여줍니다.

표 4-7 zh\_CN.GB18030 로캘용 비트맵 글꼴

패밀리 이름	하위 패밀리	형식	코드화
Song	R	PCF (12,14,16,20,24)	GB18030-2000

다음 표는 간체 한자용으로 지원되는 코드 세트 변환을 보여줍니다.

표 4-8 간체 한자용 코드 세트 변환

코드	기호	대상 코드	기호
GB2312-80	zh_CN.euc	ISO 2022-7	zh_CN.iso2022-7
GB2312-80	zh_CN.euc	ISO 2022-CN	zh_CN.iso2022-CN
GB2312-80	zh_CN.euc	UTF-8	UTF-8
GB18030	zh_CN.gb18030	UTF-8	UTF-8
HZ-GB-2312	HZ-GB-2312	GB2312-80	zh_CN.euc
HZ-GB-2312	HZ-GB-2312	GBK	zh_CN.gbk
HZ-GB-2312	HZ-GB-2312	UTF-8	UTF-8
ISO2022-7	zh_CN.iso2022-7	GB2312-80	zh_CN.euc
ISO2022-CN	zh_CN.iso2022-CN	GB2312-80	zh_CN.euc



표 4-8 간체 한자용 코드 세트 변환 (계속)

코드	기호	대상 코드	기호
ISO2022-CN	zh_CN.iso2022-CN	UTF-8	UTF-8
ISO2022-CN	zh_CN.iso2022-CN	zh.GBK	zh_CN.gbk
UTF-8	UTF-8	GB2312-80	zh_CN.euc
UTF-8	UTF-8	GB18030	zh_CN.gb18030
UTF-8	UTF-8	ISO2022-CN	zh_CN.iso2022-CN
UTF-8	UTF-8	zh.GBK	zh_CN.gbk
zh.GBK	zh_CN.gbk	ISO2022-CN	zh_CN.iso2022-CN
zh.GBK	zh_CN.gbk	UTF-8	UTF-8

## 정체 한자 지역화

Solaris 9 제품의 정체 한자는 다섯 가지 로케를 제공합니다:

- zh\_TW.EUC CNS11643.1992 코드 세트를 코드화하기 위해서는 EUC 체계가 사용됩니다.
- zh\_TW.BIG5 로케이 Big5를 지원합니다.
- zh\_TW.UTF-8 로케이 유니 코드 3.1을 지원합니다.
- zh\_HK.BIG5HK 로케이 Big5-HKSCS를 지원합니다.
- zh\_HK.UTF-8 로케이 유니 코드 3.1을 지원합니다.

정체 한자는 주로 대만 및 홍콩, 중국에서 사용됩니다. 다음 입력 방식은 zh\_TW.EUC, zh\_TW.BIG5 및 zh\_TW.UTF-8 로케에서 지원됩니다:

- 새 ChuYin
- ChuYin
- TsangChieh
- Array
- BoShiaMy
- DaYi
- JianYi
- 광둥어
- EUC NeiMa
- Big5 NeiMa
- 영어-중국어
- 별도의 코드 테이블 입력 방식(예: PinYin)
- 정체 한자용 입력 방식 보조 창 지원

다음 입력 방식은 zh\_HK.BIG5HK 및 zh\_HK.UTF-8 로케에서 지원됩니다.

- ChuYin
- TsangChieh
- Array
- BoShiaMy
- DaYi
- JianYi
- 광동어
- BIG5+HKSCS NeiMa
- 영어-중국어
- 별도의 코드 테이블 입력 방식(예: PinYin)
- 정체 한자용 입력 방식 보조 창 지원
- 새 ChuYin

다음 표는 zh\_TW 로캘용 정체 한자 트루 타입 글꼴을 보여줍니다.

표 4-9 zh\_TW 로캘용 정체 한자 트루 타입 글꼴

전체 패밀리 이름	하위 패밀리	형식	판매업체	코드화
Hei	R	트루 타입	Hanyi	CNS11643.1992
Kai	R	트루 타입	Hanyi	CNS11643.1992
Ming	R	트루 타입	Hanyi	CNS11643.1992

다음 표는 zh\_TW 로캘용 정체 한자 비트맵 글꼴을 보여줍니다.

표 4-10 zh\_TW 로캘용 정체 한자 비트맵 글꼴

전체 패밀리 이름	하위 패밀리	형식	코드화
Ming	R	PCF (12,14,16,20,24)	CNS11643.1992

다음 표는 zh\_HK.BIG5HK 로캘용 트루 타입 글꼴을 보여줍니다.

표 4-11 zh\_HK.BIG5HK 로캘용 트루 타입 글꼴

패밀리 이름	하위 패밀리	형식	판매업체	코드화
Ming	R	트루 타입	FangZheng	Big5-HKSCS
Hei	R	트루 타입	FangZheng	Big5-HKSCS
Kai	R	트루 타입	FangZheng	Big5-HKSCS

다음 표는 zh\_HK.BIG5HK 로캘용 비트맵 글꼴을 보여줍니다.

표 4-12 zh\_HK.BIG5HK 로캘용 비트맵 글꼴

패밀리 이름	하위 패밀리	형식	코드화
Ming	R	PCF(12,14,16,20,24)	Big5-HKSCS

다음 표는 정체 한자용으로 지원되는 코드 세트 변환을 보여줍니다.

표 4-13 정체 한자용 코드 세트 변환

코드	기호	대상 코드	기호
BIG5	zh_TW-big5	CNS 11643	zh_TW-euc
BIG5	zh_TW-big5	ISO2022-CN	zh_TW-iso2022-CN-EXT
BIG5	zh_TW-big5	UTF-8	UTF-8
BIG5+HKSCS	zh_HK.big5hk	UTF-8	UTF-8
CNS 11643	zh_TW-euc	BIG5	zh_TW-big5
CNS 11643	zh_TW-euc	UTF-8	UTF-8
CNS 11643	zh_TW-euc	ISO2022-7	zh_TW-iso2022-7
CNS 11643	zh_TW-euc	ISO2022-CN-EXT	zh_TW-iso2022-CN-EXT
CNS 11643	zh_TW-euc	UTF-8	UTF-8
ISO2022-7	zh_TW-iso2022-7	CNS 11643	zh_TW-euc
ISO2022-7	zh_TW-iso2022-7	UTF-8	UTF-8
ISO2022-CN	zh_TW-iso2022-CN-EXT	BIG5	zh_TW-big5
ISO2022-CN-EXT	zh_TW-iso2022-CN-EXT	CNS 11643	zh_TW-euc
UTF-8	UTF-8	BIG5	zh_TW-big5
UTF-8	UTF-8	BIG5+HKSCS	zh_HK.big5hk
UTF-8	UTF-8	CNS 11643	zh_TW-euc
UTF-8	UTF-8	ISO 2022-7	zh_TW-iso2022-7

---

## 일본어 지역화

이 절에서는 일본어 로케일 관련 정보를 설명합니다.

### 일본어 로케일

서로 다른 문자 코드화를 지원하는 네 가지 일본어 로케일을 Solaris 9 환경에서 이용할 수 있습니다. ja 및 ja\_JP.eucJP 로케일은 일본어 EUC를 바탕으로 합니다. ja\_JP.eucJP 로케일은 UI-OSF Japanese Environment Implementation Agreement 버전 1.1을, ja 로케일은 이전 Solaris 릴리스의 전통적인 사양을 준수합니다. ja\_JP.PCK 로케일은 PC-간지 코드(Shift\_JIS)를 바탕으로 하며, ja\_JP.UTF-8은 UTF-8을 바탕으로 합니다.

일본어 EUC와 문자 세트 사이의 맵은 eucJP(5) 설명서 페이지를 참조하십시오. PC-간지 코드와 문자 세트 사이의 맵은 PCK(5) 설명서 페이지를 참조하십시오.

### 일본어 문자 세트

지원되는 일본어 문자 세트는 다음과 같습니다:

- JIS X 0201-1976
- JIS X 0208-1990
- JIS X 0212-1990
- JIS X 0213-2000 (유니 코드 3.1에서 정의된 문자에 한함)

JIS X 0212-1990은 ja\_JP.PCK 로케일에서 지원되지 않습니다. JIS X 0213-2000은 ja\_JP.UTF-8 로케일에서만 지원됩니다. JIS X 0213-2000에서 정의된 모든 문자를 사용할 수 있는 것은 아닙니다. 유니 코드 3.1 문자 세트에 정의되어 있는 문자만 사용할 수 있습니다.

판매자 정의 문자(VDC) 및 사용자 정의 문자(UDC)도 지원됩니다. VDC는 JIS X 0208-1990 또는 JIS X 0212-1990의 사용되지 않은(예약된) 코드 포인트를 점유합니다. UDC는 VDC와 동일한 코드 포인트를 점유하지만 VDC용으로 할당된 코드 포인트는 제외입니다.

### 일본어 글꼴

세 가지 일본어 글꼴 형식(비트맵, 트루 타입 및 Type1)이 지원됩니다. 일본어 Type1 글꼴에는 인쇄용 JIS X 0212만 포함됩니다. Type1 글꼴은 UDC에 의해서도 사용됩니다.

일본어 비트맵 글꼴은 다음 표에서 설명합니다.

표 4-14 일본어 비트맵 글꼴

전체 패밀리 이름	하위 패밀리	형식	판매자	코드화
sun gothic	R, B	PCF(12,14,16,20,24)		JIS X 0208-1983, JIS X 0201-1976
sun minchou	R	PCF(12,14,16,20,24)		JIS X 0208-1983, JIS X 0201-1976
ricoh hg gothic b	R	PCF (10,12,14,16,18,20,24)	RICOH	JIS X 0208-1983, JIS X 0201-1976
ricoh hg mincho l	R	PCF (10,12,14,16,18,20,24)	RICOH	JIS X 0208-1983, JIS X 0201-1976
ricoh gothic	R	PCF (10,12,14,16,18,20,24)	RICOH	JIS X 0212-1990, JIS X 0213-2000
ricoh mincho	R	PCF (10,12,14,16,18,20,24)	RICOH	JIS X 0212-1990, JIS X 0213-2000
ricoh heiseimin	R	PCF(12,14,16,18,20,24)	RICOH	JIS X 0212-1990

일본어 트루 타입 글꼴은 다음 표에서 설명합니다.

표 4-15 일본어 트루 타입 글꼴

전체 패밀리 이름	하위 패밀리	형식	판매자	코드화
ricoh hg gothic b	고정	트루 타입	RICOH	JIS X 0208-1983, JIS X 0201-1976
ricoh hg mincho l	고정	트루 타입	RICOH	JIS X 0208-1983, JIS X 0201-1976
ricoh gothic	고정, 비례	트루 타입	RICOH	JIS X 0201-176, JIS X 0208-1983, JIS X 0213-2000
ricoh mincho	고정, 비례	트루 타입	RICOH	JIS X 0201-1976, JIS X 0208-1983, JIS X 0213-2000
ricoh heiseimin	고정	트루 타입	RICOH	JIS X 0212-1990

## 일본어 입력 시스템

ATOK12는 Solaris 9 환경의 기본 일본어 입력 시스템입니다. 일본어 로캘이 설치되어 있으면 모든 일본어 로캘과 UTF-8 로캘용으로 사용할 수 있습니다. Wnn6 일본어 입력 시스템도 모든 일본어 로캘에서 사용할 수 있습니다. 작업 공간 메뉴에서 입력 시스템을 전환할 수 있습니다. 일본어 Solaris 1.x BCP 지원을 위해서는 kkcj 일본어 입력 시스템을 이용할 수 있습니다.

다음 예에서는 ATOK12를 사용해 일본어를 입력하는 방법을 설명합니다.

1. Control + spacebar 를 눌러 변환 모드를 켜십시오.
2. 가나 문자 텍스트를 입력하십시오(예: `kanjihenkan`).
3. 스페이스바를 눌러 간지 문자로 변환하십시오.  
다른 간지 문자를 표시하려면 스페이스바를 눌러 변환 후보 테이블을 표시하십시오.  
선택할 번호를 입력하십시오.
4. 전체 텍스트를 간지 문자로 변환하려면 Return을 누르십시오.  
선택된 문자만 변환하려면 아래쪽 화살표를 누르십시오.
5. Control + spacebar를 눌러 변환 모드를 끄십시오.

## 일본어 단말기용 단말기 설정

문자 기반 단말기(TTY)에서 일본어 로케를 사용하려면 라인 편집이 올바르게 작동되도록 단말기를 설정해 사용해야 합니다.

- 단말기가 CDE 단말기 에뮬레이터인 경우에는 (`dtterm`) 일본어 로케(`ja`, `ja_JP.PCK` 또는 `ja_JP.UTF-8`)에서 인자 `-defeucw` 와 함께 `stty(1)`를 사용하십시오. 로케 `ja`에서의 예를 들면 다음과 같습니다:

```
% setenv LANG ja
% stty defeucw
```

- 단말기가 CDE 단말기 에뮬레이터가 아니지만 터미널의 코드 세트가 현재 로케의 코드 세트와 동일하다면 인자 `-defeucw`와 함께 `stty(1)`를 사용하십시오.
- 단말기의 코드 세트가 현재 로케의 코드 세트와 같지 않다면 `setterm(1)`을 사용해 코드 변환을 활성화시킵니다. 예를 들어, `ja` 로케에 있지만 단말기가 `PCK (Shift_JIS code)`를 필요로 하면 다음과 같이 지정하십시오:

```
% setenv LANG ja
% setterm -x PCK
```

자세한 내용은 `setterm(3CURSES)` 설명서 페이지를 참조하십시오.

## 일본어 iconv 모듈

`iconv(1)` 및 `iconv(3)`을 통해 몇 가지 일본어 코드 세트 변환이 지원됩니다. 자세한 내용은 `iconv_ja(5)` 설명서 페이지를 참조하십시오.

## 사용자 정의 문자 지원

사용자 정의 문자 유틸리티인 `sdtudctool`은 윤곽선 글꼴(Type1)과 비트맵 글꼴(PCF)을 모두 처리합니다. `fontedit`, `type3creator` 및 `fontmanager`와 같은 일부 유틸리티를 사용해 이전 릴리스의 구식 유틸리티로 작성된 UDC 글꼴을 이전할 수도 있습니다.

## 부분 로켈과 전체 로켈의 차이

다음 구성 요소는 일본어 Language CD가 있는 전체 로켈 환경에서만 사용할 수 있습니다:

- 번역된 메시지, 도움말 및 설명서 페이지
- Wnn6 일본어 입력 시스템
- 일본어 Solaris 1.x BCP 지원
- Mincho (min\*) 서체 비트맵 글꼴
- JIS X 0212 Type1 인쇄용 글꼴
- 일본어 지향 단순 프린터 및 jpostprint 지원
- kanji(1)와 같은 레거시 일본어 유틸리티

---

## 한국어 지역화

1995년 12월, 한국 정부는 ISO 10646-1/유니 코드 2.0을 바탕으로 한 표준 한국어 코드 세트인, KS X 1005-1을 발표했습니다.

ISO-10646 문자 세트는 두 가지 공통 문자 세트를 사용합니다:

- UCS-2. 범용 문자 세트 (2바이트 형식)
- UCS-4. 범용 문자 세트 (4바이트 형식)

ISO-10646 문자 세트는 IBM PC 기반 운영 체제에서 직접 사용할 수 없습니다. 예를 들어, Solaris 운영 환경의 커널과 기타 많은 모듈들은 특정 바이트 값들을 문자열의 널 문자 (0x00)와 같은 제어 명령으로 해석합니다. ISO-10646 문자 세트는 첫 번째 또는 두 번째 바이트에서 어떤 비트 조합으로든 코드화할 수 있습니다. ISO-10646 문자는 이러한 제한을 갖고 있는 Solaris 시스템에서 자유롭게 전송할 수 없습니다.

이전 경로를 확립하기 위해 ISO-10646 문자 세트는 C0 제어문자(0x00..0x1F), C1 제어문자(0x80..0x9F), 공백문자(0x20), DEL(0x7F)을 사용하지 않고서 이 ISO-10646 문자들을 다시 코드화하는 UCS 변환 형식(UTF)을 정의합니다.

ko.UTF-8은 한국어 표준 코드 세트인 KS X 1005-1을 지원하는 Solaris 로켈입니다. 이 로켈은 이전 KS X 1005 및 모두 11,172 개 한국어 문자를 지원합니다. 한국어 UTF-8은 한국어 관련 ISO-10646 문자 및 글꼴을 지원합니다. ISO-10646은 전 세계 모든 문자들을 다루기 때문에 어떤 언어든 문자에 관계 없이 입력 및 출력을 할 수 있도록 다양한 입력 방식과 글꼴들이 제공됩니다. 범용 UTF/UCS를 사용할 수 있게 되기 전까지는 한국어 UTF-8이 이전 한국어 표준 코드 세트 및 확장된 ASCII뿐 아니라 한국어 문자와 관련된 ISO-10646 코드 일부를 지원합니다.

ko 로켈에서는 EUC 체계를 사용해 KS X 1001을 코드화합니다. ko.UTF-8 로켈은 KS X 1001의 슈퍼 세트인 KS X 1005-1/유니 코드 2.0 코드 세트를 지원합니다. 이 두 가지 로켈은 최종 사용자가 보기에는 동일한 것으로 보이지만 내부 문자 코드화는 서로 다릅니다. 한국어 Solaris 제품은 다음 입력 방식을 지원합니다:

ko 로케일의 경우

- 한글 2-벌식 (자음과 모음 각 한 세트)
- 한글-한자 변환
- 특수 문자
- 16진수 코드

ko.UTF-8 로케일의 경우

- 한글 2-벌식 (자음과 모음 각 한 세트)
- 한글-한자 변환
- 특수 문자
- 16진수 코드

다음 표는 ko 로케일용 한국어 비트맵 글꼴을 보여줍니다.

표 4-16 ko 로케일용 Solaris 9 한국어 비트맵 글꼴

전체 패밀리 이름	하위 패밀리	형식	코드화
고딕	R/B	PCF (12,14,16,18,20,24)	KS X 1001
그래픽	R/B	PCF (12,14,16,18,20,24)	KS X 1001
해서	R/B	PCF (12,14,16,18,20,24)	KS X 1001
고딕	R/B	PCF (12,14,16,18,20,24)	KS X 1001
명조	R/B	PCF (12,14,16,18,20,24)	KS X 1001
필기	R/B	PCF (12,14,16,18,20,24)	KS X 1001
둥근 고딕	R/B	PCF (12,14,16,18,20,24)	KS X 1001

다음 표는 ko.UTF-8 로케일용 한국어 비트맵 글꼴을 보여줍니다.

표 4-17 ko.UTF-8 로케일용 Solaris 9 한국어 비트맵 글꼴

전체 패밀리 이름	하위 패밀리	형식	코드화
고딕	R/B	PCF (12,14,16,18,20,24)	KS X 1001 (조합)
그래픽	R/B	PCF (12,14,16,18,20,24)	KS X 1001 (조합)
해서	R/B	PCF (12,14,16,18,20,24)	KS X 1001 (조합)
고딕	R/B	PCF (12,14,16,18,20,24)	KS X 1001 (조합)
명조	R/B	PCF (12,14,16,18,20,24)	KS X 1001 (조합)
필기	R/B	PCF (12,14,16,18,20,24)	KS X 1001 (조합)

다음 표는 ko/ko.UTF-8 로케일용 한국어 트루 타입 글꼴을 보여줍니다.



표 4-18 ko/ko.UTF-8 로컬용 Solaris 9 한국어 트루 타입 글꼴

전체 패밀리 이름	하위 패밀리	형식	판매자	코드화
고딕	R	트루 타입	한양	유니 코드
명조	R	트루 타입	한양	유니 코드
해서	R	트루 타입	한양	유니 코드
동근 고딕	R	트루 타입	한양	유니 코드

다음 표는 한국어 iconv를 보여줍니다.

표 4-19 한국어 iconv

코드	기호	대상 코드	기호
IBM CP933	cp933	UTF-8 (유니코드 2.0)	ko_KR-UTF-8
ISO646	646	KS X 1001	5601
ISO2022-KR	iso2022-7	KS X 1001	ko_KR-euc
ISO2022-KR	iso2022-7	UTF-8 (유니코드 2.0)	ko_KR-UTF-8
KS X 1001	5601	UTF-8	UTF-8
KS X 1001	EUC-KR	UTF-8	UTF-8
KS X 1001	KSC5601	UTF-8	UTF-8
KS X 1001	ko_KR-euc	UTF-8 (유니코드 2.0)	ko_KR-UTF-8
KS X 1001	ko_KR-euc	ISO2022-KR	ko_KR-iso2022-7
KS X 1001	ko_KR-euc	KS X 1001	ko_KR-johap
KS X 1001	ko_KR-euc	KS X 1001	ko_KR-johap92
KS X 1001	ko_KR-euc	KS X 1001	ko_KR-nbyte
KS X 1001	ko-KR-nbyte	KS X 1001	ko_KR-euc
KS X 1001	ko-KR-johap	UTF-8 (유니코드 2.0)	ko_KR-UTF-8
KS X 1001	ko-KR-johap	KS X 1001	ko_KR-euc
KS X 1001	ko-KR-johap92	UTF-8 (유니코드 2.0)	ko_KR-UTF-8
KS X 1001	ko-KR-johap92	KS X 1001	ko_KR-euc
UTF-8	UTF-8	KS X 1001	5601
UTF-8	UTF-8	KS X 1001	EUC-KR
UTF-8	UTF-8	KS X 1001	KSC5601

표 4-19 한국어 iconv (계속)

코드	기호	대상 코드	기호
UTF-8	ko-KR-UTF-8	IBM CP 933	cp_933
UTF-8	ko-KR-UTF-8	KS X 1001	ko_KR-euc
UTF-8	ko-KR-UTF-8	ISO2022-KR	ko_KR-iso2022-7
UTF-8	ko-KR-UTF-8	KS X 1001	ko_KR-johap
UTF-8	ko-KR-UTF-8	KS X 1001	ko_KR-johap92

### UTF-8 로캘 지원의 개요

---

이 절은 다음에 대해 설명합니다.

- “Unicode의 개요” 91 페이지
- “Unicode 로캘: en\_US.UTF-8 지원 개요” 92 페이지
- “데스크탑 입력 방식” 94 페이지
- “시스템 환경” 112 페이지
- “코드 변환” 116 페이지
- “DtMail” 117 페이지
- “프로그래밍 환경” 119 페이지

---

### Unicode의 개요

Unicode 표준은 컴퓨터 처리를 위한 텍스트 표현에 사용되는 범용 문자 인코딩 표준입니다. 이것은 국제 표준 ISO/IEC 10646-1:2000 및 ISO/IEC 10646-2:2001과 완전 호환되고 ISO/IEC 10646과 동일한 모든 문자와 인코딩 포인트를 포함합니다. Unicode 표준은 문자와 문자 사용에 대한 추가 정보를 제공합니다. Unicode를 준수하는 모든 구현은 ISO/IEC 10646도 준수합니다.

Unicode 표준은 다국어 일반 텍스트 인코딩의 일관성 있는 방식을 제공하고 국제 텍스트 파일 교환을 촉진합니다. 다국어 텍스트를 사용하는 컴퓨터 사용자, 비즈니스맨, 언어학자, 연구자, 과학자 등은 Unicode 표준이 업무를 대단히 간소화한다고 생각합니다. 정기적으로 수학 기호와 기타 기술 문자를 사용하는 수학자와 기술자도 Unicode 표준을 가치있다고 여깁니다.

Unicode는, 17개의 16비트 플레인을 통해 최대 1,114,112개의 코드 포인트를 지원할 수 있습니다. 각 플레인은 65,536개의 다른 코드 포인트를 가집니다.

Unicode를 지원하는 백만개 이상의 코드 포인트 중에서, Unicode 버전 3.1은 현재 94,140개의 문자를 플레인 0, 1, 2 및 14에 정의해 놓았습니다. 플레인 15와 16은 사용자 정의 문자라고도 하며, 사용자 전용 목적을 위해 존재합니다. 플레인 15와 16은 총 131,068개 까지의 사용자 정의 문자를 지원할 수 있습니다.

Unicode는 다음 문자 인코딩 체계를 사용하여 인코딩될 수 있습니다.

- UTF-8
- UTF-16
- UTF-32

UTF-8은 ASCII 문자 코드 값을 그대로 보존하는, Unicode 가변폭 인코딩 방식입니다. 이 형식은 Solaris Unicode 로켈에서 파일을 저장하는 방식으로 쓰입니다.

UTF-16은 Unicode의 16비트 인코딩 형식입니다. UTF-16에서는, 65,536까지의 문자는 하나의 16비트 값으로 인코딩됩니다. 65,536 이상부터 1,114,111까지의 코드 포인트에 해당되는 문자들은 두 개의 16비트 값이 한 쌍이 되어 인코딩됩니다.(surrogates라고도 합니다.)

UTF-32는 Unicode의 고정폭, 21비트 인코딩 방식으로 대개 32비트 저장 또는 데이터 유형으로 표현됩니다. 이 형식은 Solaris Unicode 로켈에서 프로세스 코드(와이드 캐릭터 코드)로 사용됩니다.

Unicode 표준 및 ISO/IEC 10646과 다양한 표현 형식에 대한 자세한 내용은 다음을 참조하십시오.

- *The Unicode Standard, Version 3.0, The Unicode Standard Annex #19: UTF-32, The Unicode Standard Annex #27: Version 3.1 Unicode* 컨소시엄
- ISO/IEC 10646-1:2000, Information Technology-Universal Multiple-Octet Character Set (UCS) - Part 1: Architecture and Basic Multilingual Plane
- ISO/IEC 10646-2: Information Technology-Universal Multiple-Octet Character Set (UCS) - Part 2: Secondary Multilingual Plane for Scripts and Symbols, Supplementary Plane for CJK Ideographs, Special Purpose Plane
- Unicode 컨소시엄 웹 사이트는 <http://www.unicode.org/>입니다.

---

## Unicode 로켈: en\_US.UTF-8 지원 개요

Unicode/UTF-8 로켈은 Unicode 3.1을 지원합니다. en\_US.UTF-8 로켈은 UTF-8을 코드 세트에 사용하여 다중 스크립트 처리 지원을 제공합니다. 이 로켈은 다중 스크립트의 텍스트 입출력을 지원하며, Solaris 운영 환경에서 이 기능을 가진 첫번째 로켈이었습니다. 다른 UTF-8 로켈의 기능은 en\_us.UTF-8의 기능과 유사합니다. 다음 en\_US.UTF-8의 논의는 이러한 로켈과 동일하게 적용됩니다.

---

주 - UTF-8은 1992년에 X/Open-Uniform Joint Internationalization Working Group (XoJIG)이 고안하고 1996년에 ISO 및 IEC가 ISO/IEC 10646-1:1993에 대한 개정 2로 승인한 파일 시스템 안전 범용 문자 세트 변환 형식입니다. 이 표준은 Unicode 컨소시엄, 국제 표준 기구 및 국제 전자 유틸리티 위원회가 Unicode 2.0 및 ISO/IEC 10646-1의 일부로 채택하였습니다.

---

Solaris의 Unicode 로케일은 Unicode 3.1 및 ISO/IEC 10646-1과 10646-2에서 정의된 모든 코드 포인트 값의 처리를 지원합니다. 지원되는 스크립트는 범 유럽권 스크립트와 아시아권 스크립트뿐 아니라 아랍어, 헤브루어, 힌디어 및 태국어와 같은 복잡한 텍스트 레이아웃 스크립트도 포함합니다. 제한된 글꼴 자원으로 인해 Solaris 9 소프트웨어는 다음 문자 세트의 문자 모양만 포함합니다.

- ISO 8859-1 (영어, 프랑스어, 스페인어 및 독일어와 같은 대부분의 서부 유럽권 언어)
- ISO 8859-2 (체코어, 폴란드어, 헝가리어와 같은 대부분의 중앙 유럽권 언어)
- ISO 8859-4 (스칸디나비아어 및 발트어)
- ISO 8859-5 (러시아어)
- ISO 8859-6 (아랍어, 더 많은 표시 형식 문자 모양을 포함)
- ISO 8859-7 (그리스어)
- ISO 8859-8 (헤브루어)
- ISO 8859-9 (터키어)
- TIS 620.2533 (태국어, 더 많은 표시 형태 문자 모양을 포함)
- ISO 8859-15 (유럽 기호가 있는 대부분의 서부 유럽권 언어)
- GB 2312-1980 (간체 한자)
- JIS X 0201-1976, JIS X 0208-1990 (일본어)
- KS X 1001: 1992 부속서 3 (한국어)
- GB 18030 (간체 한자)
- HKSCS (정체 한자, 홍콩)
- Big5 (정체 한자, 대만)
- ISCII라고도 하는 IS 13194.1991(힌디어, 더 많은 표시 형식 문자 모양 포함)

만약 en\_US.UTF-8 로케일에서, 해당 글리프가 없는 문자를 보려고 하면, 로케일은 아래와 같이 "no-glyph"라는 메시지를 표시합니다.



로켈은 설치시 선택 가능하며 시스템 기본 로켈로 지정될 수 있습니다.

64비트와 32비트의 Solaris 시스템에서, en\_US.UTF-8과 동일한 수준의 로켈 지원이 제공됩니다.

---

주 - Motif 및 CDE 데스크탑 응용프로그램과 라이브러리는 en\_US.UTF-8 로켈을 지원합니다. 그러나, XView™ 및 OLIT 라이브러리는 en\_US.UTF-8 로켈을 지원하지 않습니다.

---

## 데스크탑 입력 방식

CDE는 Xm Toolkit을 사용하여 국제화된 응용프로그램에 지역화된 입력을 입력하는 기능을 제공합니다. XmText [Field] 위젯트는 각 로켈의 입력 방식과의 인터페이스가 가능합니다. 어떤 언어 환경에서는 텍스트를 오른쪽에서 왼쪽으로, 위에서 아래로 등등의 방식으로 쓰기 때문에 입력 방식을 국제화합니다. 동일한 응용프로그램 내에서 여러 글꼴을 적용하는 다른 입력 방식을 사용할 수 있습니다.

사전 편집 영역은 사전 편집 중인 문자열을 표시합니다. 이것은 네 가지 모드로 이루어질 수 있습니다

- OffTheSpot
- OverTheSpot (기본값)
- 루트
- 없음

OffTheSpot 모드에서의 위치는 상태 영역의 오른쪽에 있는 기본 창 영역 바로 아래입니다. OverTheSpot 모드에서 사전 편집 영역은 커서 포인트에 있습니다. 루트 모드에서 사전 편집 및 상태 영역은 클라이언트 창과 떨어져 있습니다.

자세한 내용은 VendorShell(3X) 설명서 페이지의 XmNpreeditType 자원 설명을 참조하십시오.

---

주 - Solaris 9 환경에서 원 아시아권 입력 방식은 간체/정체 한자, 일본어 및 한국어에 대하여 존재합니다. 이러한 메소드는 Unicode 로켈의 현재 다중 스크립트 입력 방식에 추가됩니다.

---

다음 논의는 선택된 입력 방식의 설명, 사용 방법 및 메소드간 전환 방법을 포함합니다.

## 스크립트 선택 및 입력 모드

Solaris Unicode 로켈은 다중 스크립트를 지원합니다. 모든 Unicode 로켈에는 총 14개의 입력 모드가 있습니다.

- 영어/유럽어
- 키릴 문자
- 그리스어
- 아라비아어
- 헤브루어
- 태국어
- 일본어
- 한국어
- 간체 한자
- 정체 한자
- 정체 한자(홍콩)
- 힌디어
- 유니 코드 16진 및 8진코드 입력 방법
- 테이블 조회 입력 방식

## 입력 방식 액세스하기

Compose 키 조합 또는 입력 모드 선택 창을 통해 특정 입력 모드로 전환이 가능합니다. 입력 모드 선택 창에 액세스하려면 응용프로그램 창의 맨아래 왼쪽 모서리에 있는 상태 영역에서 마우스 버튼 1을 누릅니다. 입력 모드 선택 창은 다음 그림과 같습니다.

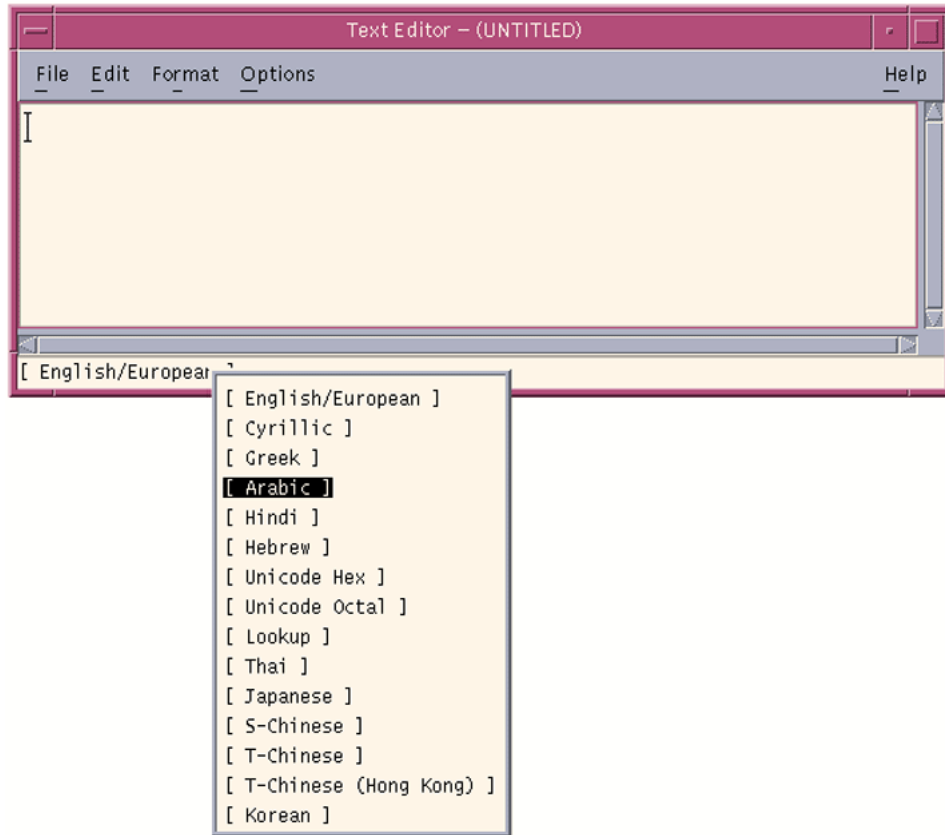


그림 5-1 입력 모드 선택 창

## 입력 모드 스위치 키 시퀀스

표 5-1에 나열된 키 시퀀스를 사용하여 현재 입력 모드를 새 입력 모드로 변경할 수 있습니다. 키 시퀀스 사용에 대한 유일한 제약은 아시아어 입력 모드 중 하나에 있을 경우 **Control**과 스페이스바를 동시에 눌러 영어/유럽어 입력 모드로 다시 전환해야 하는 것입니다. 영어/유럽어 입력 모드에 있으면 키 시퀀스를 사용하여 다른 입력 모드로 자유롭게 전환할 수 있습니다.

다음 키 시퀀스는 영어/유럽어 입력 모드에서 키릴 문자로 전환하는 방법을 보여줍니다.

- **Compose** 키를 누릅니다.
- **c** 키를 눌렀다 놓습니다.
- **c**를 누릅니다.



표 5-1 입력 모드 스위치 키 시퀀스

키 시퀀스	입력 모드
Control + 스페이스바	영어/유럽어
Compose c c	키릴 문자
Compose g g	그리스어
Compose a r	아라비아어
Compose h h	헤브루어
Compose t t	태국어
Compose h i	힌디어
Compose j a	일본어
Compose k o	한국어
Compose s c	간체 한자
Compose t c	정체 한자
Compose h k	정체 한자(홍콩)
Compose u o	유니 코드 8진 코드 입력 방법
Compose u h	유니 코드 16진 코드 입력 방법
Compose l l	테이블 조회 입력 방식

## 영어/유럽어 입력 모드

영어/유럽어 입력 모드는 영어 알파벳과 구별 표시가 있는 문자(예를 들어, á, è, î, ô, ù)와 유럽어 스크립트의 문자(예를 들어, ÿ, Œ, ç)를 포함합니다.

이 입력 모드는 모든 응용프로그램에 대한 기본 모드입니다. 입력 모드는 GUI 응용프로그램 창의 맨아래 왼쪽 모서리에서 표시됩니다.

구별 표시가 있는 문자 또는 라틴-1, 라틴-2, 라틴-3, 라틴-4, 라틴-5 및 라틴-6의 특수 문자를 삽입하려면 다음 예제의 설명과 같이 Compose 키 시퀀스를 입력해야 합니다.

Ä 문자를 표시하려면:

- Compose 키를 눌렀다 놓습니다.
- Shift와 A를 동시에 누릅니다. Shift와 A를 놓습니다.
- "를 눌렀다 놓습니다.

ç, 문자를 표시하려면:

- Compose 키를 눌렀다 놓습니다.
- ?를 눌렀다 놓습니다.

- ?를 눌렀다 놓습니다.

키보드에 사용 가능한 Compose 키가 없으면 Control 키와 Shift 키를 동시에 눌러 해당 작업을 에플레이트할 수 있습니다.

로캘에서 유럽 통화 기호(Unicode 값 U+20AC)의 입력을 위해서는 다음 입력 시퀀스 중 하나를 사용할 수 있습니다.

- AltGraph를 누른채 E 키를 함께 누름
- AltGraph를 누른채 4 키를 함께 누름
- AltGraph를 누른채 5 키를 함께 누름

이러한 입력 시퀀스로 두 키를 동시에 누릅니다. 키보드에 사용 가능한 AltGraph 키가 없으면 Alt 키로 대신할 수 있습니다.

다음 표는 Solaris 운영 환경에 대하여 가장 공통적으로 사용되는 라틴-1, 라틴-2, 라틴-3, 라틴-4, 라틴-5 및 라틴-9의 작성 시퀀스를 보여줍니다.

다음 표는 공통 라틴-1 Compose 키 시퀀스를 나열합니다.

표 5-2 공통 라틴-1 Compose 키 시퀀스

Compose 키를 누른 후, 아래 키를 눌렀다 놓음	아래 키를 눌렀다 놓습니다.	결과
스페이스바	스페이스바	중단 없는 공간
s	1	위첨자 1
s	2	위첨자 2
s	3	위첨자 3
!	!	전치 느낌표
x	o	통화 기호 ¢
p	!	문단 기호
/	u	mu u
,	"	예음 악센트
,	, (침표)	세디유
"	"	분음 기호
-	^	장음 기호
o	o	각도 °
x	x	곱셈 부호 x
+	-	플러스-마이너스
-	-	짧은 하이픈-

표 5-2 공통 라틴-1 Compose 키 시퀀스 (계속)

Compose 키를 누른 후, 아래 키를 눌렀다 놓음	아래 키를 눌렀다 놓습니다.	결과
-	:	분리 부호÷
-	a	서수(여성) <sup>a</sup>
-	o	서수(남성) <sup>a</sup>
-	, (침표)	부호 없음
.	.	가운데 점
1	2	상분수
1	4	상분수
3	4	상분수
<	<	왼쪽 이중 꺾쇠 인용 표시
>	>	오른쪽 이중 꺾쇠 인용 표시
?	?	전치 물음표
A	˘ (반대쪽 인용)	억음 기호 À
A	´ (단일 인용)	예음 Á
A	*	위쪽 링
A	"	분음 부호 Ä
A	^	Â곡절 부호
A	~	틸데
A	E	AE 복모음 Æ
C	, (침표)	C 세디유 Ç
C	o	저작권 기호
D	-	대문자 eth ð
E	˘ (반대쪽 인용)	E 억음 기호 È
E	´	E 예음 기호 É
E	"	E 분음 기호 ë
E	^	E 곡절 기호
I	˘ (반대쪽 인용)	I 억음 기호 Ì
I	´	I 예음 기호 Í
I	"	I 분음 기호 ï

표 5-2 공통 라틴-1 Compose 키 시퀀스 (계속)

Compose 키를 누른 후, 아래 키를 눌렀다 놓음	아래 키를 눌렀다 놓습니다.	결과
I	^	I 곡절 기호
L	-	파운드 기호 £
N	~	N 틸데
O	˘ (반대쪽 인용)	O 억음 기호 Ò
O	'	O 예음 기호 Ó
O	/	O 슬래시 Ø
O	"	O 분음 기호 Ö
O	^	O 곡절 기호 Ô
O	~	O 틸데
R	o	등록 상표
T	H	손
U	˘ (반대쪽 인용)	U 억음 기호 Û
U	'	U 예음 기호 Ú
U	"	U 분음 기호 Ü
U	^	U 곡절 기호 Û
Y	'	Y 예음 기호 ý
Y	-	엔 기호 ¥
a	˘ (반대쪽 인용)	a 억음 기호 à
a	'	a 예음 기호 á
a	*	a 위쪽 링ă
a	"	a 분음 기호
a	~	a 틸데
a	^	a 곡절 기호
a	e	ae 복모음 Æ
c	, (침표)	c 세디유
c	/	센트 기호 ¢
c	o	저작권 기호
d	-	eth ð

표 5-2 공통 라틴-1 Compose 키 시퀀스 (계속)

Compose 키를 누른 후, 아래 키를 눌렀다 놓음	아래 키를 눌렀다 놓습니다.	결과
e	˘ (반대쪽 인용)	e 억음 기호
e	'	e 예음 기호
e	"	e 분음 기호
e	^	e 곡절 ê 기호
i	˘ (반대쪽 인용)	i 억음 기호 ì
i	'	i 예음 기호 í
i	"	i 분음 기호 Ï
i	^	i 곡절 기호 î
n	~	n 틸데
o	˘ (반대쪽 인용)	o 억음 기호 ò
o	'	o 예음 기호 ó
o	/	o 슬래시 ø
o	"	o 분음 기호 Ö
o	^	o 곡절 기호 ô
o	~	o 틸데 õ
s	s	독일어 이 중 s는 경음 S라고도 함
t	h	손
u	˘ (반대쪽 인용)	u 억음 기호 ù
u	'	u 예음 기호 ú
u	"	u 분음 기호 Ü
u	^	u 곡절 기호 û
y	'	y 예음 기호 ý
y	"	y 분음 기호 ÿ
		깨진 표시줄 !

다음 표는 공통 라틴-2 및 라틴-4 Compose 키 시퀀스를 나열합니다.

표 5-3 공통 라틴-2 Compose 키 시퀀스

Compose 키를 누른 후, 아래 키를 눌렀다 놓음	아래 키를 눌렀다 놓습니다.	결과
a	스페이스바	ogonek
u	스페이스바	약음 기호
v	스페이스바	caron
"	스페이스바	이중 예음 기호
A	a	A ogonek
A	u	A 약음 기호
C	'	C 예음 기호
C	v	C caron
D	v	D caron
-	D	D 스트로크
E	v	E caron
E	a	E ogonek
L	'	L 예음 기호
L	-	L 스트로크
L	>	L caron
N	'	N 예음 기호
N	v	N caron
O	>	O 이중 예음 기호
S	'	S 예음 기호
S	v	S caron
S	,	S 세디유
R	'	R 예음 기호
R	v	R caron1
T	v	T caron
T	,	T 세디유
U	*	U 위쪽 링ă
U	>	U 이중 예음 기호
Z	'	Z 예음 기호

표 5-3 공통 라틴-2 Compose 키 시퀀스 (계속)

Compose 키를 누른 후, 아래 키를 눌렀다 놓음	아래 키를 눌렀다 놓습니다.	결과
Z	v	Z caron
Z	.	Z 위쪽 점
k	k	kra
A	_	A 장음 기호
E	_	E 장음 기호
E	.	E 위쪽 점
G	,	G 세디유
Y	_	I 장음 기호
Y	~	I 탈데
Y	a	I ogonek
K	,	K 세디유
L	,	L 세디유
N	,	N 세디유
O	_	O 장음 기호
R	,	R 세디유
T		T 스트로크
U	~	U 탈데
U	a	U ogonek
U	_	U 장음 기호
N	N	Eng
a	_	a 장음 기호
e	_	e 장음 기호
e	.	e 위쪽 점
g	,	g 세디유
i	_	i 장음 기호
i	~	i 탈데
i	a	i ogonek1
k	,	k 세디유

표 5-3 공통 라틴-2 Compose 키 시퀀스 (계속)

Compose 키를 누른 후, 아래 키를 눌렀다 놓음	아래 키를 눌렀다 놓습니다.	결과
l	,	l 세디유
n	,	n 세디유
o	_	o 장음 기호
r	,	r 세디유
t		t 스트로크
E	~	u 틸데
E	a	u ogonek
E	_	E 장음 기호
n	n	eng

다음 표는 공통 라틴-3 Compose 키 시퀀스를 나열합니다.

표 5-4 공통 라틴-3 Compose 키 시퀀스

Compose 키를 누른 후, 아래 키를 눌렀다 놓음	아래 키를 눌렀다 놓습니다.	결과
C	>	C 곡절 기호
C	.	C 위쪽 점
G	>	G 곡절 기호
G	.	G 위쪽 점
H	>	H 곡절 기호
J	>	j 곡절 기호
S	>	S 곡절 기호
U	u	U 약음 기호
c	>	c 곡절 기호
c	.	c 위쪽 점
g	>	g 곡절 기호
g	.	g 위쪽 점
h	>	h 곡절 기호



표 5-4 공통 라틴-3 Compose 키 시퀀스 (계속)

Compose 키를 누른 후, 아래 키를 눌렀다 놓음	아래 키를 눌렀다 놓습니다.	결과
j	>	j 곡절 기호
s	>	s 곡절 기호
u	u	u 약음 기호

다음 표는 공통 라틴-5 Compose 키 시퀀스를 나열합니다.

표 5-5 Common 라틴-5 Compose 키 시퀀스

Compose 키를 누른 후, 아래 키를 눌렀다 놓음	아래 키를 눌렀다 놓습니다.	결과
G	u	G 약음 기호
I	.	I 위쪽 점
g	u	g 약음 기호
i	.	i 점 없음

다음 표는 공통 라틴-9 Compose 키 시퀀스를 나열합니다.

표 5-6 공통 라틴-9 Compose 키 시퀀스

Compose 키를 누른 후, 아래 키를 눌렀다 놓음	아래 키를 눌렀다 놓습니다.	결과
o	e	슬러 oe
O	E	슬러 OE
Y	"	Y 분음 기호

## 아랍어 입력 모드

아랍어 입력 모드로 전환하려면 Compose a r을 입력하거나 입력 모드 선택 창에서 아랍어를 선택합니다. 입력 모드 선택 창 액세스에 대한 자세한 내용은 그림 5-1을 참조하십시오.

다음 그림은 아랍어 키보드 레이아웃을 보여줍니다.

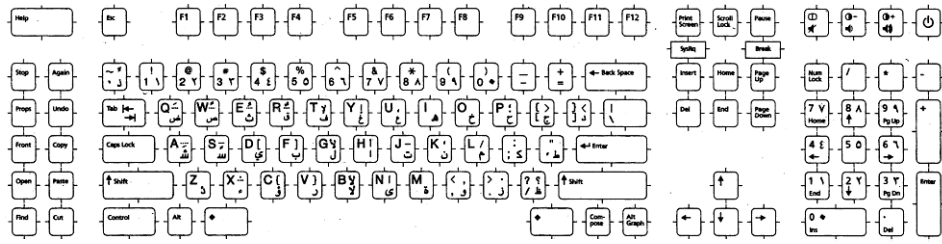


그림 5-2 아랍어 키보드

## 키릴 문자 입력 모드

키릴 문자 입력 모드로 전환하려면 Compose c c 를 누르거나 입력 모드 선택 창에서 키릴 문자를 선택합니다. 입력 모드 선택 창 액세스에 대한 자세한 내용은 그림 5-1을 참조하십시오.

키릴 문자(러시아어) 키보드 레이아웃이 다음 그림에 나타납니다.

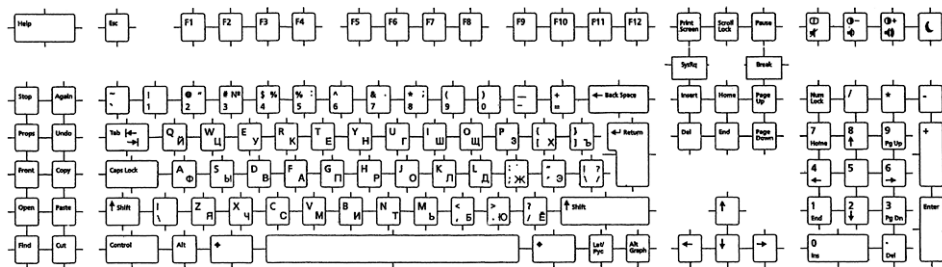


그림 5-3 키릴 문자(러시아어) 키보드

키릴 문자 입력 모드로 전환한 후 영어 또는 유럽어 텍스트를 입력할 수 없습니다. 영어/유럽어 입력 모드로 다시 전환하려면 Control과 스페이스바를 동시에 누르거나 상태 영역을 눌러 입력 모드 선택 창에서 영어/유럽어 입력 모드를 선택합니다. 그림 5-1을 참조하십시오.

해당 입력 모드 전환 키 시퀀스를 입력하여 다른 입력 모드로 전환할 수도 있습니다.

## 그리스어 입력 모드

그리스어 입력 모드로 전환하려면 Compose g g를 누르거나 입력 모드 선택 창에서 그리스어를 선택합니다. 입력 모드 선택 창 액세스에 대한 자세한 내용은 그림 5-1을 참조하십시오.

그리스어 입력 모드로 전환한 후 영어 또는 유럽어 텍스트를 입력할 수 없습니다. 영어/유럽어 입력 모드로 다시 전환하려면 Control과 스페이스바를 동시에 누르거나 상태 영역을 눌러 입력 모드 선택 창에서 영어/유럽어 입력 모드를 선택합니다. 그리스어 유럽 키보드 레이아웃이 다음 그림에 나타납니다.

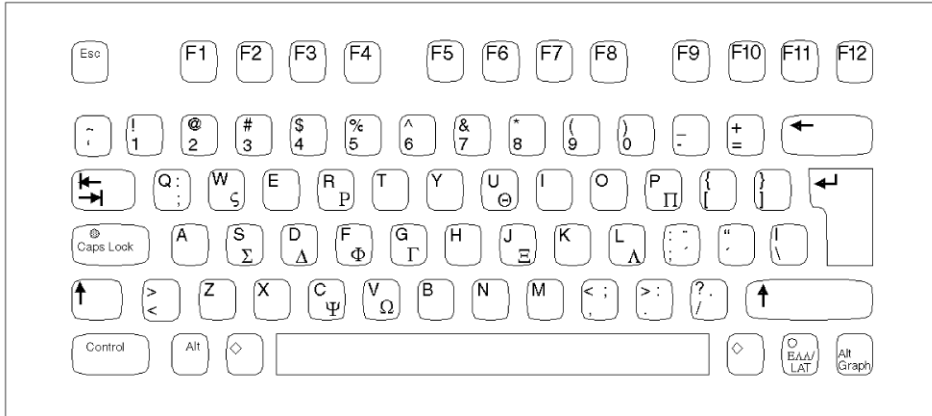


그림 5-4 그리스어 유럽 키보드

다음 그림은 그리스어 UNIX 키보드를 보여줍니다.

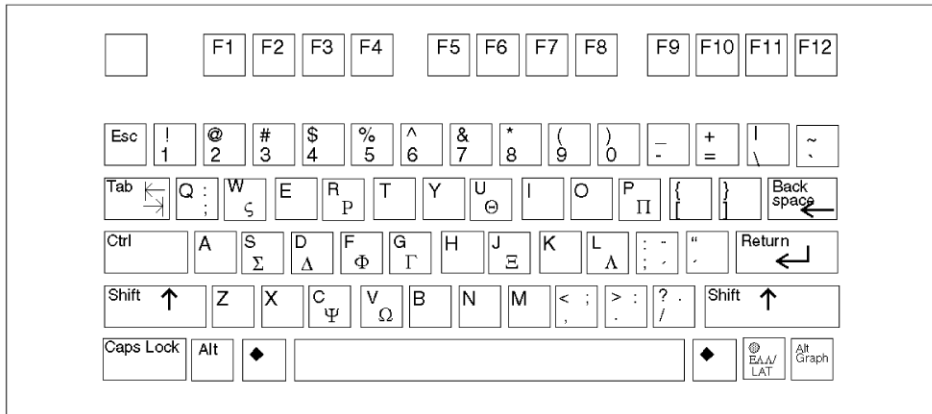


그림 5-5 그리스어 UNIX 키보드

## 헤브루어 입력 모드

헤브루어 입력 모드로 전환하려면 Compose h h 를 누르거나 입력 모드 선택 창에서 헤브루어를 선택합니다. 입력 모드 선택 창 액세스에 대한 자세한 내용은 그림 5-1을 참조하십시오.

다음 그림은 헤브루어 키보드 레이아웃을 보여줍니다.



그림 5-6 헤브루어 키보드

## 힌디어 입력 모드

힌디어 입력 모드로 전환하려면 Compose 키를 누르거나 입력 모드 선택 창에서 힌디어를 선택합니다. 입력 모드 선택 창 액세스에 대한 자세한 내용은 그림 5-1을 참조하십시오. 영어/유럽어 입력 모드로 다시 전환하려면 Control과 스페이스바를 동시에 누르거나 상태 영역을 눌러 입력 모드 선택 창에서 영어/유럽어 입력 모드를 선택합니다.



그림 5-7 힌디어 키보드

힌디어-Shift 키보드 레이아웃은 다음 그림과 같습니다.



그림 5-8 힌디어-Shift 키보드

## 일본어 입력 모드

일본어 입력 모드로 전환하려면 Compose ja 를 누르거나 입력 모드 선택 창에서 일본어를 선택합니다. 입력 모드 선택 창 액세스에 대한 자세한 내용은 그림 5-1을 참조하십시오.

일본어 입력 시스템을 사용하려면 하나 이상의 일본어 로케일을 설치하고 시스템을 재부팅해야 합니다. 일본어 로케일을 설치한 후 모든 UTF-8 로케일에서 ATOK12를 사용할 수 있습니다. Wnn6은 ja\_JP.UTF-8을 제외한 UTF-8 로케일에서 사용할 수 없습니다.

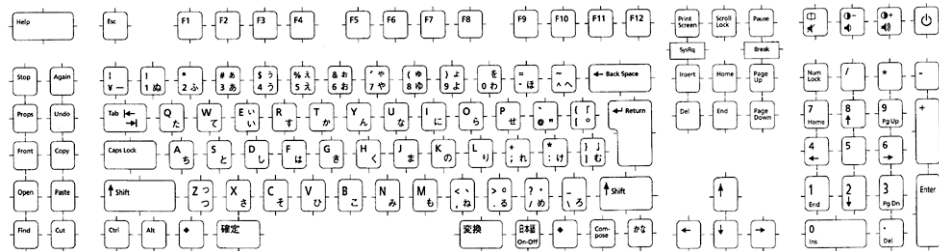


그림 5-9 일본어 키보드

## 한국어 입력 모드

한국어 입력 모드로 전환하려면 Compose ko 를 누르거나 입력 모드 선택 창에서 한국어를 선택합니다. 입력 모드 선택 창 액세스에 대한 자세한 내용은 그림 5-1을 참조하십시오.

한국어 입력 시스템에 액세스하려면 시스템에 하나 이상의 한국어 로캘을 설치해야 합니다. 한국어 로캘을 설치하면 한국어 입력 시스템을 사용할 수 있습니다. 한국어 입력 시스템 사용 방법에 대한 자세한 내용은 *Korean Solaris User's Guide*를 참조하십시오.

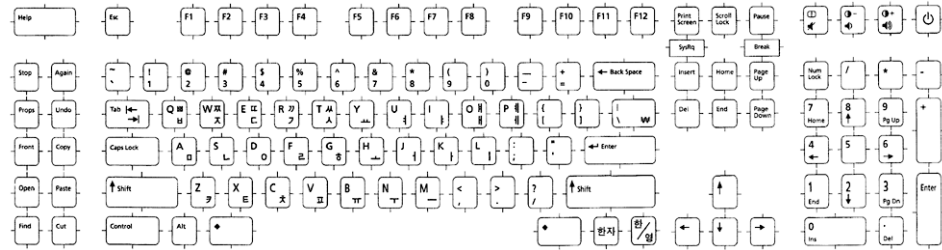


그림 5-10 한국어 키보드

## 간체 한자 입력 모드

간체 한자 입력 모드로 전환하려면 Compose s c 를 누르거나 입력 모드 선택 창에서 간체 한자를 선택합니다. 입력 모드 선택 창 액세스에 대한 자세한 내용은 그림 5-1을 참조하십시오.

간체 한자 입력 시스템을 사용하려면 시스템에 하나 이상의 간체 한자 로캘을 설치해야 합니다. 간체 한자 입력 시스템 사용 방법에 대한 자세한 내용은 *Simplified Chinese Solaris User's Guide*를 참조하십시오.

## 태국어 입력 모드

태국어 입력 모드로 전환하려면 Compose t t를 누르거나 입력 모드 선택 창에서 태국어를 선택합니다. 입력 모드 선택 창 액세스에 대한 자세한 내용은 그림 5-1을 참조하십시오.

영어/유럽어 입력 모드로 다시 전환하려면 Control과 스페이스바를 동시에 누르거나 상태 영역을 눌러 입력 모드 선택 창에서 영어/유럽어 입력 모드를 선택합니다.

다음 그림은 태국어 키보드 레이아웃을 보여줍니다.

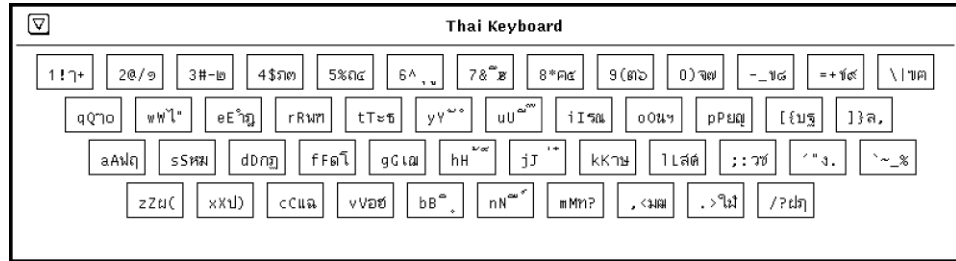


그림 5-11 태국어 키보드

## 정체 한자 입력 모드

정체 한자 입력 모드로 전환하려면 Compose t c 를 누르거나 입력 모드 선택 창에서 정체 한자를 선택합니다. 입력 모드 선택 창 액세스에 대한 자세한 내용은 그림 5-1을 참조하십시오.

원 정체 한자 입력 시스템에 액세스하려면 시스템에 하나 이상의 정체 한자 로케를 설치해야 합니다. 정체 한자 입력 시스템 사용 방법에 대한 자세한 내용은 *Traditional Chinese Solaris User's Guide*를 참조하십시오.

## 정체 한자(홍콩) 입력 모드

정체 한자 입력 모드로 전환하려면 Compose h c 를 누르거나 입력 모드 선택 창에서 정체 한자(홍콩)를 선택합니다. 입력 모드 선택 창 액세스에 대한 자세한 내용은 그림 5-1을 참조하십시오.

원 정체 한자(홍콩) 입력 시스템에 액세스하려면 시스템에 하나 이상의 정체 한자(홍콩) 로케를 설치해야 합니다.

## Unicode 16진법 입력 모드

Unicode 16진법 입력 모드로 전환하려면 Compose u h 를 누르거나 입력 모드 선택 창에서 Unicode 16진법을 선택합니다. 8진법 숫자 시스템으로 전환하려면 Compose u o 를 누르거나 Unicode 8진법을 선택합니다. 입력 모드 선택 창 액세스에 대한 자세한 내용은 그림 5-1을 참조하십시오.

입력 모드를 사용하려면, 문자의 16진법 또는 8진법 코드 포인트 값을 알아야 합니다. 코드 포인트 값과 문자 사이의 매핑에 대해서는 *The Unicode Standard, Version 3.0*을 참조하십시오.

Unicode 16진법 코드 입력 모드에서 문자를 입력하려면 4 개의 16진법 숫자를 입력합니다. 일부 예제의 16진법 값은 다음과 같습니다.

- 반전 느낌표는 00A1
- 그리스어 작은 문자 베타는 03B2
- 한국어 한글 음절은 AC00
- 일본어 가타카나 문자 A는 30A1 A
- 통합 한 문자는 4E58

16진법 숫자에 대하여 A, B, C, D, E, F의 대소문자 모두를 사용할 수 있습니다. 16진법 대신 8진법 숫자 시스템을 선호하면 0~7의 8진법 숫자를 입력할 수 있습니다. 숫자 하나 또는 둘을 잘못 입력하면 Delete 또는 백스페이스 키를 사용하여 숫자를 삭제할 수 있습니다.

## 테이블 조회 입력 모드

테이블 조회 입력 모드로 전환하려면 Compose i i 를 누르거나 입력 모드 선택 창에서 조회를 선택합니다. 입력 모드 선택 창 액세스에 대한 자세한 내용은 그림 5-1을 참조하십시오.

두번째 조회 창이 그룹 전용 표시에 대한 문자를 한 번에 최대 8개 보여줍니다. 다음 후보 세트에 대해서는 Control n을 누르고 이전 후보 세트에 대해서는 Control p를 누릅니다.

---

## 시스템 환경

이 절은 로켈 환경 변수, TTY 환경 설치, 32비트 및 64비트 STREAMS 모듈, 단말기 지원을 설명합니다.

### 로켈 환경 변수

시스템에 en\_US.UTF-8 로켈이 설치되어 있는지 확인하십시오. 변수 범주의 현재 로켈 설정을 확인하려면 로켈(1) 유틸리티를 사용하십시오.

```
system% locale
LANG=en_US.UTF-8
LC_CTYPE="en_US.UTF-8"
LC_NUMERIC="en_US.UTF-8"
LC_TIME="en_US.UTF-8"
LC_COLLATE="en_US.UTF-8"
LC_MONETARY="en_US.UTF-8"
LC_MESSAGES="en_US.UTF-8"
LC_ALL=
```

en\_US.UTF-8 로켈 데스크탑 환경을 사용하려면 로켈을 먼저 선택하십시오. TTY 환경에서 다음 C-셸 예제에서와 같이 LANG 환경 변수를 en\_US.UTF-8 로 설정하여 먼저 로켈을 선택하십시오.



```
system% setenv LANG en_US.UTF-8
```

LC\_ALL, LC\_COLLATE, LC\_CTYPE, LC\_MESSAGES, LC\_NUMERIC, LC\_MONETARY 및 LC\_TIME 범주는 설정되지 않거나 en\_US.UTF-8로 설정됩니다. 범주가 설정되면 보다 낮은 우선 순위의 LANG 환경 변수를 무시합니다. 환경 변수의 계층에 대한 자세한 내용은 setlocale(3C) 설명서 페이지를 참조하십시오.

CDE 데스크탑에서 en\_US.UTF-8 환경을 시작할 수도 있습니다. CDE 로그인 화면의 Options -> Language 메뉴에서 en\_US.UTF-8을 선택하십시오.

## TTY 환경 설정

사용 중인 단말기 및 단말기 에뮬레이터에 따라 특정한 코드 세트 특정 STREAMS 모듈을 스트림에 푸시해야 할 수 있습니다.

STREAMS 모듈 및 일반적인 스트림에 대한 자세한 내용은 *STREAMS Programming Guide*를 참조하십시오.

다음 표는 단말기 환경에서 en\_US.UTF-8 로캘이 지원하는 STREAMS 모듈을 보여줍니다.

표 5-7 en\_US.UTF-8이 지원하는 32비트 STREAMS 모듈

32비트 STREAMS 모듈	설명
/usr/kernel/strmod/u8lat1	UTF-8과 ISO8859-1 (서부 유럽어) 사이의 코드 변환 STREAMS 모듈
/usr/kernel/strmod/u8lat2	UTF-8과 ISO8859-2 (동부 유럽어) 사이의 코드 변환 STREAMS 모듈
/usr/kernel/strmod/u8koi8	UTF-8과 KOI8-R (키릴 문자) 사이의 STREAMS 모듈

다음 표는 en\_US.UTF-8이 지원하는 64비트 STREAMS 모듈을 나열합니다.

표 5-8 en\_US.UTF-8이 지원하는 64비트 STREAMS 모듈

64비트 STREAMS 모듈	설명
/usr/kernel/strmod/sparcv9/u8lat1	UTF-8과 ISO8859-1(서부 유럽어) 사이의 코드 변환 STREAMS 모듈
/usr/kernel/strmod/sparcv9/u8lat2	UTF-8과 ISO8859-2(동부 유럽어) 사이의 코드 변환 STREAMS 모듈
/usr/kernel/strmod/sparcv9/u8koi8	UTF-8과 KOI8-R(키릴 문자) 사이의 코드 변환 STREAMS 모듈

## 커널에서 STREAMS 모듈 로드하기

커널에서 STREAMS 모듈을 로드하려면 먼저 루트가 되어야 합니다.

64비트 Solaris 또는 32비트 Solaris 시스템 실행 여부를 확인하려면 다음과 같이 `isainfo(1)` 유틸리티를 사용하십시오.

```
system# isainfo -v
64-bit sparcv9 applications
32-bit sparcv9 applications
```

명령이 이 정보를 반환하면 64비트 Solaris 시스템을 실행 중입니다. 32비트 Solaris 시스템을 실행 중이면 유틸리티가 다음을 표시합니다.

```
system# isainfo -v
32-bit sparcv9 applications
```

시스템이 STREAMS 모듈에 이미 로드되지 않았음을 확인하려면 `modinfo(1M)`을 사용하십시오.

```
system# modinfo | grep modulename
```

`u8lat1`과 같은 STREAMS 모듈이 이미 설치되었으면 출력은 다음과 같습니다.

```
system# modinfo | grep u8lat1
89 ff798000 4b13 18 1 u8lat1 (UTF-8 <--> ISO 8859-1 module)
```

모듈이 이미 설치되었으면 로드할 필요 없습니다. 그러나 모듈이 아직 로드되지 않았으면 다음과 같이 `modload(1M)`을 사용하십시오.

```
system# modload /usr/kernel/strmod/u8lat1
```

이 명령은 커널에 32비트 `u8lat1` STREAMS 모듈을 로드하여 스트림으로 푸시할 수 있습니다. 64비트 Solaris 제품을 실행 중이면 다음과 같이 `modload(1M)`을 사용하십시오.

```
system# modload /usr/kernel/strmod/sparcv9/u8lat1
```

STREAMS 모듈이 커널에 로드되고 이제 스트림으로 푸시할 수 있습니다.

커널에서 모듈을 언로드하려면 다음과 같이 `modunload(1M)`을 사용하십시오. 이 예제에서 `u8lat1` 모듈이 언로드됩니다.

```
system# modinfo | grep u8lat1
89 ff798000 4b13 18 1 u8lat1 (UTF-8 <--> ISO 8859-1 module)
system# modunload -i 89
```

## dtterm 및 UTF-8 문자의 입출력이 가능한 단말기

Solaris 운영 환경의 이전 릴리스와 달리 `dtterm(1)` 단말기와 UTF-8 코드 세트의 입출력을 지원하는 다른 단말기는 스트림에 추가 STREAMS 모듈이 필요 없습니다. `ldterm(7M)` 모듈은 이제 코드 세트 독립이며 Unicode/UTF-8도 지원합니다.

Unicode 로케일의 적합한 단말기 환경을 설정하려면 `stty(1)` 유틸리티를 사용하십시오. 현재 설정을 질의하려면 다음과 같이 `stty(1)` 유틸리티의 `-a` 옵션을 사용하십시오.

```
system% /bin/stty -a
```

---

주 - `/usr/ucb/stty`는 국제화되지 않았기 때문에 대신 `/bin/stty`를 사용하십시오.

---

## 라틴-1, 라틴-2 또는 KOI8-R에 대한 단말기 지원

라틴-1 (ISO8859-1), 라틴-2 (ISO8859-2) 또는 KOI8-R만 지원하는 단말기의 경우 다음 STREAMS 구성이 있어야 합니다.

```
head <-> ttcompat <-> ldterm <-> u8lat1 <-> TTY
```

이 구성은 라틴-1을 지원하는 단말기 전용입니다. 라틴-2 단말기에 대하여 STREAMS 모듈 `u8lat1`을 `u8lat2`로 바꾸십시오. KOI8-R 단말기에 대하여 모듈을 `u8koi8`로 바꾸십시오.

커널로 이미 STREAMS 모듈을 로드했는지 확인하십시오.

위와 같이 STREAMS 구성을 설정하려면 예제의 두번째 명령줄에서와 같이 `strchg(1M)`를 사용하십시오.

```
system% cat> tmp/mystreams
ttcompat
ldterm
u8lat1
ptem
^D
system% strchg -f /tmp/mystreams
```

`strchg(1)`를 사용할 때 장치의 루트 또는 소유자인지 확인하십시오. 현재 구성을 확인하려면 다음과 같이 `strconf(1)`를 사용하십시오.

```
system% strconf
ttcompat
ldterm
u8lat1
ptem
pts
system%
```

원래 구성을 재설정하려면 다음과 같이 STREAMS 구성을 설정하십시오.

```
system% cat> /tmp/orgstreams
ttcompat
ldterm
ptem
^D
system% strchg -f /tmp/orgstreams
```

## ~/.cshrc에 설정 저장하기

필요한 STREAMS 모듈이 커널로 이미 로드되었다고 가정하면 편의적으로 .cshrc 파일(C 셸 예제)에 다음 행을 저장할 수 있습니다.

```
setenv LANG en_US.UTF-8
if ($?USER != 0 && $?prompt != 0) then
    cat>! /tmp/mystreams$$ << _EOF
    ttcompat
    ldterm
    u8lat1
    ptem
_EOF
    /bin/strchg -f /tmp/mystreams$$
    /bin/rm -f /tmp/mystreams$$
    /bin/stty cs8 -istrip defeucw
endif
```

이러한 행이 .cshrc 파일에 있으면 STREAMS 모듈을 사용할 때마다 모든 명령을 입력할 필요가 없습니다. 두번째 \_EOF가 파일의 첫번째 열에서 시작해야 함을 주의하십시오.

---

## 코드 변환

Unicode 로케일 지원은 iconv(1), iconv(3C)와 sdtconvtool(1)을 통해 많은 국가의 주요 코드 세트 사이의 다양한 코드 변환을 추가합니다.

Solaris 9 환경에서 유틸리티 geniconvtbl을 사용하여 사용자 정의 코드 변환이 가능합니다. geniconvtbl 유틸리티를 사용하여 작성된 사용자 정의 코드 변환은 iconv(1) 및 iconv(3) 모두와 함께 사용될 수 있습니다. 이 유틸리티에 대한 자세한 내용은 geniconvtbl(1) 및 geniconvtbl(4) 설명서 페이지를 참조하십시오.

iconv(1), iconv\_open(3C) 및 sdtconvtool(1)에 적용할 수 있는 사용 가능한 fromcode 및 tocode 이름이 부록 A의 표에서 표시됩니다. iconv 코드 변환에 대한 자세한 내용은 iconv(1), iconv\_open(3C), iconv(3), iconv\_close(3C), geniconvtbl(1), geniconvtbl(4) 및 sdtconvtool(1) 설명서 페이지를 참조하십시오. 사용 가능한 코드 변환에 대한 자세한 내용은 iconv\_en\_US.UTF-8(5), iconv(5), iconv\_ja(5), iconv\_ko(5), iconv\_zh(5) 및 iconv\_zh\_TW(5) 설명서 페이지를 참조하십시오. 부록 A도 참조하십시오.

---

주 - UCS-2, UCS-4, UTF-16과 UTF-32는, 문자 스트림의 시작이 Unicode 3.1과 ISO/IEC 10646-1:2000에서 정의한 BOM(Byte Order Mark)일 경우, 이 문자를 인식하는 Unicode/ ISO/IEC 10646의 형식입니다. UCS-2BE, UCS-4BE, UTF-16BE, UTF-32BE와 같은 다른 형식은 모두 BOM 문자를 인식하지 않는 고정 너비 Unicode/ISO/IEC 10646 표현 형식이고 또한 빅 엔디언 바이트 순서를 가정합니다. 반면 UCS-2LE, UCS-4LE, UTF-16LE, UTF-32LE 같은 표현 형식은 리틀 엔디언 바이트 정렬을 가정합니다. 또한 BOM 문자를 인식하지 않습니다.

ISO8859-\* 및 KOI8-\*의 관련 스크립트 및 언어에 대해서는 <http://czyborra.com/charsets/iso8869.html>을 참조하십시오.

---

---

## DtMail

스크립트의 늘어난 적용 범위의 결과로 en\_US.UTF-8 로케에서 실행되는 Solaris 9 DtMail은 MIME이름이 나타내는 다음 문자 세트를 지원합니다.

- US-ASCII (7비트 US ASCII)
- UTF-8 (UCS 전송 형식 8비트)
- UTF-7 (UCS 전송 형식 7비트)
- ISO-8859-1 (라틴-1)
- ISO-8859-2 (라틴-2)
- ISO-8859-3 (라틴-3)
- ISO-8859-4 (라틴-4)
- ISO-8859-5 (라틴/키릴 문자)
- ISO-8859-6 (라틴/아랍어)
- ISO-8859-7 (라틴/그리스어)
- ISO-8859-8 (라틴/헤브루어)
- ISO-8859-9 (라틴-5)
- ISO-8859-10 (라틴-6)
- ISO-8859-13 (라틴-7/발트어)
- ISO-8859-14 (라틴-8/켈트어)
- ISO-8859-15 (라틴-9)
- ISO-8859-16 (라틴-10)
- KOI8-R (키릴 문자)
- ISO-2022-JP 및 EUC-JP (일본어)
- ISO-2022-KR 및 EUC-KR (한국어)
- ISO-2022-CN (간체 한자)
- ISO-8859-13 (라틴-7/발트어)
- ISO-8859-14 (라틴-8/켈트어)
- KOI8-U (키릴 문자/우크라이나어)
- Shift\_JIS (Shift JIS의 일본어)
- GB2312 (EUC의 간체 한자)

- TIS-620 (태국어)
- UTF-16 (UCS 전송 형식 16비트)
- UTF-16BE (UTF-16 빅 엔디언)
- UTF-16LE (UTF-16 리틀 엔디언)
- Windows-1250
- Windows-1251
- Windows-1252
- Windows-1253
- Windows-1254
- Windows-1255
- Windows-1256
- Windows-1257
- Windows-1258
- Big5 (정체 한자)
- UTF-16 (UCS 전송 형식 32비트)
- UTF-32BE (UTF-32 빅 엔디언)
- UTF-32LE (UTF-32 리틀 엔디언)

이 지원은 사용자가 DtMail의 단일 인스턴스로 세계 모든 지역의 다양한 문자 세트로 인코딩된 모든 종류의 전자 우편을 볼 수 있도록 지원합니다. DtMail은 전자 우편과 함께 제공되는 MIME 문자 세트와 내용 전송 인코딩을 찾아 수신된 전자 우편을 암호 해독합니다. Windows-125x MIME 문자 세트가 지원됩니다.

전자 우편을 보내는 경우 수신자 우편 사용자 에이전트(우편 클라이언트)가 이해하는 MIME 문자 세트를 지정해야 하고 그렇지 않으면 en\_US.UTF-8 로케일이 제공하는 기본 MIME 문자 세트를 사용할 수 있습니다. 새 메시지 창에서 Control Y를 누르거나 형식 메뉴 버튼을 누른 다음 문자 세트 변경 버튼을 눌러 보내는 전자 우편의 문자 세트를 전환할 수 있습니다. 다음 사용 가능 문자 세트 이름이 보내기 버튼의 맨 위 왼쪽 모서리에 표시됩니다.

전자 우편 메시지 헤더나 메시지 본문이 지정된 MIME 문자 세트가 표현할 수 없는 문자를 포함하면 시스템이 자동으로 모든 문자를 표현할 수 있는 UTF-8로 문자 세트를 전환합니다.

메시지가 7비트 US-ASCII 문자 세트의 문자만 포함하면 전자 우편의 기본 MIME 문자 세트는 US-ASCII입니다. 모든 우편 사용자 에이전트는 문자 또는 정보의 상실 없이 그러한 전자 우편 메시지를 해석할 수 있습니다.

메시지가 스크립트 혼합의 문자를 포함하면 기본 MIME 문자 세트는 UTF-8입니다. UTF-8의 모든 8비트 문자는 할당된 인쇄 가능 인코딩으로 인코딩됩니다. MIME, 등록된 MIME 문자 세트, 할당된 인쇄 가능 인코딩에 대한 자세한 내용은 RFCs 2045, 2046, 2047, 2048, 2049, 2279, 2152, 2237, 1922, 1557, 1555 및 1489를 참조하십시오.



그림 5-12 DtMail 새 메시지 창

## 프로그래밍 환경

국제화된 응용프로그램은 자동으로 en\_US.UTF-8 로케일을 사용해야 합니다. 그러나 응용프로그램 자원 파일의 적합한 FontSet/XmFontList 정의가 필수입니다.

국제화된 응용프로그램에 대한 자세한 내용은 *Creating Worldwide Software: Solaris International Developer's Guide, 2nd edition*을 참조하십시오.

## X 응용프로그램과 함께 사용되는 FontSet

X 응용프로그램과 함께 사용되는 FontSet에 대한 자세한 내용은 “Unicode 로케일: en\_US.UTF-8 지원 개요” 92 페이지를 참조하십시오.

Solaris 9 환경은 CDE 데스크탑 환경을 지원하기 때문에 각 문자는 보증된 글꼴 세트를 갖습니다.

다음은 Solaris 9 제품에서 지원되는 라틴-1 글꼴입니다.

```
-dt-interface system-medium-r-normal-xxs sans utf-10-100-72-72-p-59-iso8859-1
-dt-interface system-medium-r-normal-xs sans utf-12-120-72-72-p-71-iso8859-1
-dt-interface system-medium-r-normal-s sans utf-14-140-72-72-p-82-iso8859-1
-dt-interface system-medium-r-normal-m sans utf-17-170-72-72-p-97-iso8859-1
-dt-interface system-medium-r-normal-l sans utf-18-180-72-72-p-106-iso8859-1
-dt-interface system-medium-r-normal-xl sans utf-20-200-72-72-p-114-iso8859-1
-dt-interface system-medium-r-normal-xxl sans utf-24-240-72-72-p-137-iso8859-1
```

-dt-interface user-\* 및 -dt-application-\* 별명을 포함하는 CDE 공통 글꼴 별명에 대한 자세한 설명은 *Common Desktop Environment: Internationalization Programmer's Guide*를 참조하십시오.

en\_US.UTF-8 로케일에서 utf도 X 논리적 글꼴 설명 이름의 스타일 필드의 추가 속성으로 로케일의 공통 글꼴 별명에 포함됩니다. 따라서 적합한 글꼴 세트를 가지려면 다음 예제에서와 같이 추가 스타일이 글꼴 세트 생성에 포함되어야 합니다.

```
fs = XCreateFontSet(display,
"-dt-interface system-medium-r-normal-s*utf*",
&missing_ptr, &missing_count, &def_string);
```

## CDE/Motif 응용프로그램의 글꼴 목록 정의

FontSet 정의에서와 같이 응용프로그램의 XmFontList 자원 정의는 로케일이 지원하는 추가 스타일 속성도 포함해야 합니다.

```
*fontList:\
-dt-interface system-medium-r-normal-s*utf*:
```



# Complex Text Layout

---

Complex Text Layout(CTL) 확장은 Motif API가 아랍어, 히브리어 및 타이어에서 필요한 것과 같이 논리적 및 물리적 텍스트 표현 사이의 복잡한 변환을 필요로 하는 쓰기 시스템을 지원할 수 있도록 합니다. CTL Motif는 슬러, 구별 기호, 세그먼트 정렬과 같은 문자 모양을 제공하며 정적 및 동적 텍스트 위젯트의 변환을 지원합니다. 또한 오른쪽에서 왼쪽 및 왼쪽에서 오른쪽 텍스트 방향과 동적 텍스트 위젯트 탭을 지원합니다. 텍스트 렌더링은 표현 레이어를 통해 처리되기 때문에 다른 위젯트 라이브러리는 CTL을 지원하도록 쉽게 확장될 수 있습니다.

---

## CTL 기술의 개요

새 기능을 활용하려면 사용자는 (PLS) 라이브러리 및 해당 언어 엔진이 있어야 합니다. CTL은 언어 엔진에 대한 인터페이스로 PLS를 사용하고 언어 엔진을 사용하여 텍스트를 렌더링하기 전에 텍스트를 변환합니다. CTL을 지원하는 응용프로그램은 CTL 설명서의 설명과 같이 추가 자원을 포함해야 합니다.

특히, XomCTL은 기본 로캘 종속 PLS 모듈 변환이 제공하는 다음과 같은 복잡한 언어 모양과 재정렬 기능을 지원합니다.

- 위치 변화
- 슬러 (다대일) 및 문자 작성(일대다)
- 구별 기호
- 양방향성
- 대칭적 교체
- 숫자 모양
- 문자열 검증

---

## CTL 구조의 개요

CTL 구조는 그림 6-1과 같이 구성됩니다. 스택 상부에 있는 Dt Apps는 텍스트 렌더링을 위해 Motif CTL 기능을 사용합니다. Motif는 PLS를 사용하여 로캘 특정 언어 엔진과 인터페이스를 이루고 위치 변화, 숫자 모양 등을 지원하는 변환을 수행합니다.

CTL 구조는 새로운 로캘 특정 엔진을 추가하여 새 언어를 지원하도록 구축됩니다. 다시 말해서, 타이어와 베트남어에 대한 지원은 Motif 또는 Dt Apps의 변경 없이 추가될 수 있습니다.

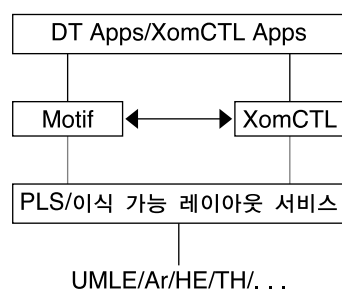


그림 6-1 CTL 구조

---

## X 라이브러리 기반 응용프로그램에 대한 CTL 지원

XomCTL(X 라이브러리 출력 모듈의 Complex Text Layout 지원)은 모든 순수 X 윈도우 응용프로그램(X 기반 단말기 에뮬레이터)이 CTL 지원을 갖도록 합니다. XomCTL은 X11 단순 글꼴 지원을 포함하는 완전 기능 개방형 소스 XI18N 구현을 제공합니다.

---

## New XOC 자원

다음 XOC 자원은 Solaris 9 환경이 제공합니다.

XNText	사용자가 CTL 작업이 수행되어야 하는 텍스트 버퍼를 설정할 수 있도록 합니다.
XNTextLayoutNumGlyphs	텍스트 버퍼에 있는 텍스트의 글리프 수를 알려줍니다.

XNTextLayoutModifier	Motif의 XmNLayoutModifier와 동일합니다.
XNTextLayoutProperty	PLS 특성, 입력 대 출력 및 출력 대 입력과 동일합니다.
XNTextLayoutMapInpToOut	PLS 특성, 입력 대 출력 및 출력 대 입력과 동일합니다.
XNTextLayoutMapOutToInp	PLS 특성, 입력 대 출력 및 출력 대 입력과 동일합니다.

이에 대한 설명은 X/Open 또는 PLS 이식 가능 서비스의 사양에서 얻을 수 있습니다.

---

## CTL 기술 지원을 위한 Motif 변경 사항

Motif의 다음 변경 사항은 CTL 기술을 지원합니다:

XmNlayoutDirection	객체 레이아웃을 제어합니다.
XmStringDirection	시스템이 문자열의 문자들을 표현하는 방향을 지정합니다.
XmRendition	XmRendition에 가상 자원을 추가합니다.
XmText 및 XmTextField	XmRendition과 관련된 텍스트의 레이아웃 동작에 영향을 줍니다.
XmTextFieldGetLayoutModifier	rendition 레이아웃 객체의 레이아웃 수정자 문자열을 반환합니다.
XmTextGetLayoutModifier	위젯트와 관련된 rendition의 현재 레이아웃 객체 설정 값을 반환합니다.
XmTextFieldSetLayoutModifier	rendition에 결합된 레이아웃 객체용 레이아웃 수정자 값을 설정합니다.
XmTextSetLayoutModifier	위젯트와 관련된 rendition의 레이아웃 객체 설정을 수정합니다.
XmStringDirectionCreate	복합 문자열을 작성합니다.

---

## XmNlayoutDirection

XmNlayoutDirection 자원은<sup>1</sup> 객체 레이아웃을 제어합니다. 아래와 같은 방식으로 LayoutObject의 방향 값과 상호 작용합니다.

### 레이아웃 방향 결정하기

XmNlayoutDirection이 XmDEFAULT\_DIRECTION으로 지정되면 위젯의 레이아웃 방향은 이를 제어하는 의사 XOC에서 작성시에 설정됩니다. 동적 텍스트(XmText 및 XmTextField)의 경우 관련 의사 XOC는 위젯용으로 사용되는 XmRendition과 관련된 것입니다. 정적 텍스트(XmList, XmLabel, XmLabelG)의 경우 레이아웃 방향은 방향을 지정하는 첫 번째 복합 문자열에서 설정됩니다. 지정은 두 가지 중 한 가지 방법으로 수행됩니다:

**직접 지정** 구성 요소가 XmSTRING\_COMPONENT\_LAYOUT\_PUSH 또는 XmSTRING\_COMPONENT\_DIRECTION 유형입니다.

**간접 지정** 구성 요소 유형이 XmSTRING\_COMPONENT\_LOCALE\_TEXT, XmSTRING\_COMPONENT\_WIDECHAR\_TEXT 또는 구성 요소의 관련 XmRendition 및 관련 LayoutObject의 XmSTRING\_COMPONENT\_TEXT입니다.

XmNlayoutDirection이 XmDEFAULT\_DIRECTION으로 지정되어 있지 않고 XmNlayoutModifier @ls orientation 값이 레이아웃 수정자 문자열에 명시적으로 지정되어 있지 않으면 XmNlayoutDirection 값이 XOC 및 그 LayoutObject를 통해 전달됩니다.

XmNlayoutDirection 및 XmNlayoutModifier @ls orientation 값이 명시적으로 지정되어 있으면 동작이 혼합됩니다. XmNlayoutDirection은 위젯 객체 레이아웃을 제어하며 XmNlayoutModifier @ls orientation 값은 레이아웃 변환을 제어합니다.

*CAE Specification: Portable Layout Services: Context-dependent and Directional Text*를 참조하십시오. Open Group: Feb 1997; ISBN 1-85912-142-X; 기존 POSIX 로케일 모델에 대한 논리적 확장 모델로서 문맥에 종속적이고 양방향인 텍스트 변환을 처리하기 위한 이식 가능한 기능들에 대해 설명하는 문서 번호 C616 설명서는 복합 텍스트 언어에 대한 지원을 제공하고자 하는 시스템 및 응용프로그램 프로그래머들을 위한 것입니다.

---

<sup>1</sup> XmNlayoutDirection 개요, 특히 XmStringDirection 및 XmNlayoutDirection 간의 상호 작용에 대한 설명은 *Motif Programmer's Guide* (Release 2.1)의 11.3절을 참조하십시오.

---

## XmStringDirection

XmStringDirection은 시스템이 문자열의 문자들을 표시하는 방향을 지정하기 위해 사용되는 데이터 유형입니다.

XmNlayoutDirection 자원은 방향을 지정하는 구성 요소를 갖고 있지 않은 임의의 복합 문자열(XmString)에 대한 기본 렌더링 방향을 설정합니다. 따라서 레이아웃 방향을 설정하려면 XmNlayoutDirection 자원을 위한 적절한 값을 설정해야 합니다. 특정 방향 구성 요소를 사용해 복합 문자열을 작성할 필요는 없습니다. 응용프로그램이 XmString을 렌더링하면 해당 응용프로그램은 방향이 명시된 문자열이 작성되었는지 확인합니다(XmStringDirection). 아무런 방향 구성 요소도 없으면 응용프로그램은 현재 위젯을 위한 XmNlayoutDirection 자원의 값을 확인하고 해당 값을 XmString을 위한 기본 렌더링 방향으로 사용합니다.

---

## XmRendition

CTL은 새 의사 자원을 다음 표에 나열되어 있는 XmRendition에 추가합니다:

표 6-1 XmRendition의 새로운 자원

이름	클래스/유형	액세스	기본값
XmNfontType	XmCFontType/XmFontType	CSG	XmAS_IS
XmNlayoutAttrObject	XmClayoutAttrObject/String	CG	NULL
XmNlayoutModifier	XmClayoutModifier/String	CSG	NULL

### XmNfontType

Rendition 글꼴 객체의 유형을 지원합니다. CTL의 경우, 자원의 값은 XmFONT\_IS\_XOC 값이어야 합니다. 값이 일치하지 않으면 XmNlayoutAttrObject 및 XmNlayoutModifier 자원은 무시됩니다.

자원의 값이 XmFont\_IS\_XOC이고 XmNfont 자원이 지정되어 있지 않으면 작성할 때 XmNfontName 자원의 값이 XmNlayoutAttrObject 자원에 의해 지정된 로케일 또는 현재 로케일의 XOC 객체로 변환됩니다. 아울러 XmNlayoutModifier 자원의 값은 XOC와 관련된 임의의 레이아웃 객체로 전달됩니다.

### XmNlayoutAttrObject

레이아웃 AttrObject 인자를 지정합니다. 자원은 XmRendition과 관련되어 있는 XOC와 관련된 레이아웃 객체를 작성하는데 사용됩니다. 문자열의 구문과 의미에 대

해서는 레이아웃 서비스 `m_create_layout()` 사양을 참조하십시오. 레이아웃 수정자 방향 출력값과 `XmNlayoutDirection` 위젯트 자원 간의 상호 작용에 대한 설명은 `XmNfontType` 설명을 참조하십시오.

#### XmNlayoutModifier

`XmRendition` 용으로 `XOC`와 함께 사용된 레이아웃 객체에 전달될 레이아웃 값을 지정합니다. 문자열의 구문과 의미에 대한 내용은 *CAE Specification*을 참조하십시오.

`XmRendition{Retrieve, Update}`을 사용해 자원을 설정하면 문자열이 `rendition`과 관련된 `XOC`와 관련된 레이아웃 객체에 전달됩니다. 이것은 레이아웃 서비스를 동적으로 구성하기 위한 체계입니다. `Orientation`, `Context`, `TypeOfText`, `TextShaping` 또는 `ShapeCharset`이 변경되면 예측할 수 없는 동작이 발생할 수 있습니다.

## 추가 레이아웃 동작

`XmNlayoutModifier`는 `XmRendition`과 관련된 텍스트의 레이아웃 동작에 영향을 줍니다. 예를 들어, 레이아웃 기본 숫자 처리가 `NUMERALS_NOMINAL`이면 사용자가 `XmNlayoutModifier` 을 `@ls numerals=nominal:national` 또는 `@ls numerals=:national`로 설정하여 `NUMERALS_NATIONAL`로 변경할 수 있습니다.

레이아웃 값은 다음 그룹으로 구분할 수 있습니다:

- 코드화 설명-`TypeOfText`, `TextShaping`, `ShapeCharset` (및 로캘 코드 세트)  
`TypeOfText`는 기본적으로 정렬을 세그먼트화하고 불투명한 블록으로 그려질 수 있습니다. 이 값을 `rendition` 객체를 통해 동적으로 수정하는 것은 대개 불필요한 일이며 예측하지 못한 동작을 유발할 수 있습니다.
- 레이아웃 동작-`Orientation`, `Context`, `ImplicitAlg`, `Swapping` 및 `Numerals`  
. `Orientation` 및 `Context`는 동적으로 수정해선 안 됩니다. `ImplicitAlg`, `Swapping` 및 `Numerals`는 안전하게 수정할 수 있습니다.
- 동작 편집하기-`CheckMode`

---

## XmText 및 XmTextField

`Xm CTL`은 시각적으로 작동하며 `Motif 2.0 CStext` 위젯트의 패턴을 따른 움직임 및 삭제 활동의 병렬 세트를 추가함으로써 `XmText` 및 `XmTextField`를 확장합니다. 표준 `Motif 2.1 Text` 및 `TextField`는 논리적 순서와 물리적 순서를 구분하지 않습니다. `next` 및 `forward`는 “오른쪽으로”를 의미하는 반면, `previous` 및 `backward`는 “왼쪽으로”를 의미합니다. 하지만 `CStext`는 올바르게 구분하여 정확한 물리적 이름(예: `left-character()`, `delete-right-word()`)으로 새 활동 세트를 정의합니다. 이 활동 루틴 모두는 위젯트의 `XmNlayoutDirection`에 민감하게 정의되며 적절한 다음 또는 이전 활동을

호출합니다. Xm CTL 확장은 CSText 확장보다 다소 복잡합니다. Xm CTL 확장은 중립적 고정화를 포함한 의사 XOC에 의한 결정에 따라 위젯트의 전체 방향보다 커서를 둘러싼 물리적 문자의 구체적 방향에 민감합니다.

또한 rendition 태그를 제공하고 배열을 제어하기 위한 선택 정책에 대한 새로운 자원도 있습니다.

새로운 Xm CTL 활동 세트는 대략 {Move, Delete, Kill}, {Left, Right}, {Character, Word}의 조합으로 이루어지며 아래에 나열되어 있습니다.

표 6-2 Xm CTL의 새로운 자원

이름	클래스/유형	액세스	기본값
XmNrenditionTag	XmCRenditionTag/XmRString	CSG	XmFONTLIST_DEFAULT_TAG
XmNalignment	XmCAlignment/XmRAlignment	CSG	XmALIGNMENT_BEGINNING
XmNeditPolicy	XmCEditPolicy/XmREditPolicy	CSG	XmEDIT_LOGICAL

**XmNrenditionTag**  
위젯트를 위해 사용되는 XmNrenderTable 자원에 있는 XmRendition의 rendition 태그를 지정합니다.

**XmNalignment**  
위젯트에서 사용되는 텍스트 배열을 지정합니다. XmALIGNMENT\_END 및 XmALIGNMENT\_CENTER만 지원됩니다.

**XmNeditPolicy**  
위젯트용으로 사용되는 편집 정책을 XmEDIT\_LOGICAL 또는 XmEDIT\_VISUAL로 지정합니다. XmEDIT\_VISUAL의 경우에는 선택, 커서 이동 및 삭제가 시각적 스타일로 표시됩니다. 자원을 설정하면 표준 키보드 이동 및 삭제 이벤트에 대한 번역도 새 “시각적” 활동 목록이나 기존 논리적 활동으로 변경됩니다.

## 문자 방향 활동 루틴

forward-cell() 및 backward-cell() 활동은 지정된 방향의 문자 방향을 질의합니다. 방향이 왼쪽에서 오른쪽 방향이면 해당하는 next-/forward- 또는 previous-/backward- 변수를 호출합니다.

## 문자 방향 추가 동작

활동은 레이아웃 서비스 변환 OutToInp 및 Property 버퍼(중첩 수준용)를 사용해 문자의 방향을 결정합니다. 따라서 위젯트의 동작은 로캘 지향 변환에 따라 달라집니다. OutToInp의 정보 또는 특히, Property 버퍼가 정확하지 않으면 위젯트는 예기치 않은 동작이 일어날 수 있습니다. 아울러, 로캘 지향 모듈이 본 사양의 범위를 벗어나면 양방향 편집 동작은 동일한 텍스트, 응용프로그램, 자원 값 및 LayoutObject 구성의 경우에도 플랫폼에 따라 차이가 있을 수 있습니다.

시각적 모드 활동은 셀 기반 동작의 표시입니다. 논리적 모드 활동은 논리적 문자 기반 동작을 유발할 수 있습니다. 예를 들어, `delete-right-character()` 작업은 디스플레이 셀에 해당하는 입력 버퍼 문자들을 삭제합니다. 즉, `LayoutObject` 변환 “속성” 바이트 “새 셀 표시 기호”가 1인 한 개의 입력 버퍼 문자 및 “새 셀 표시 기호”가 0인 뒤따르는 모든 문자입니다.<sup>2</sup>

마찬가지로 `backward-character()`의 경우 삽입점은 입력 버퍼에서 한 문자 뒤로 이동되며 커서는 관련된 출력 버퍼 문자에 해당하는 시각적 위치로 복귀합니다. 이것은 복합적인 디스플레이 셀을 가로질러 이동하려면 여러 번 키 입력을 해야 함을 의미합니다; 커서는 삽입점이 “새 셀 표시 기호”가 0인 (즉, 분음 부호 또는 조각 묶음 기호) 입력 버퍼 문자를 가로질러 이동할 때 디스플레이 위치는 실제로는 변경되지 않습니다.

이것은 삭제 작업이 논리/입력 버퍼 측이나 물리적/출력측의 디스플레이 셀 수준에서 일어난다는 것을 의미합니다. 입력 및 출력 버퍼 간의 1 대 1 대응이 성립되지 않기 때문에 엄격한 물리적 문자 대 문자 삭제를 위한 모드는 없습니다. 주어진 물리적 문자는 예를 들어 논리적 문자의 한 단편만을 나타낼 수 있습니다.

## XmText 활동 루틴

다음 목록은 XmText 활동 루틴을 설명합니다.

### `left-character(extend)`

`XmNeditPolicy`가 `XmEDIT_LOGICAL`이고 인자가 없이 호출되면 삽입 커서가 논리적으로 한 문자 뒤로 이동합니다. 삽입 커서가 줄의 시작 부분에 있으면 삽입 커서는 이전 줄의 논리적인 마지막 문자로 이동합니다. 그렇지 않으면 삽입 커서 위치는 변경되지 않습니다.

`XmNeditPolicy`가 `XmEDIT_VISUAL`이면 커서는 커서 위치의 왼쪽으로 이동합니다. 삽입 커서가 줄의 시작 부분에 있으면 삽입 커서는 이전 줄의 끝 문자로 이동합니다.

`left-character()`가 인자를 통해 호출되면 `extend` 인자가 없는 경우와 마찬가지로 삽입 커서가 이동하고 현재 선택이 확장됩니다.

`left-character()` 활동은 합리적인 값 `XmCR_MOVING_INSERT_CURSOR`을 가진 `XmNmotionVerifyCallback` 프로시저에 대한 호출을 생성합니다. `extend` 인자를 통해 호출하면 이 활동은 `XmNgainPrimaryCallback` 프로시저에 대한 호출을 생성할 수 있습니다. 자세한 정보는 *Motif Programmer's Reference*의 콜백 설명을 참조하십시오.

### `right-character(extend)`

`XmNeditPolicy`가 `XmEDIT_LOGICAL`이고 아무런 인자가 없이 호출되면 삽입 커서가 논리적으로 한 문자 앞으로 이동합니다. 삽입 커서가 줄의 논리적인 끝 부분에 있으면 삽입 커서는 다음 줄의 논리적인 시작 부분으로 이동합니다.

---

<sup>2</sup> Property 버퍼에 대한 자세한 정보는 *CAE Specification*의 `m_transform_layout()` 사양을 참조하십시오.



XmNeditPolicy가 XmEDIT\_VISUAL이면 커서는 커서 위치의 오른쪽으로 이동합니다. 삽입 커서가 줄의 끝 부분에 있으면 삽입 커서는 다음 줄의 시작 부분으로 이동합니다.

extend 인자를 통해 호출되면 인자가 없는 경우와 마찬가지로 삽입 커서를 이동시키며 현재 선택을 확장합니다.

right-character() 활동은 적절한 값 XmCR\_MOVING\_INSERT\_CURSOR으로 XmNmotionVerifyCallback 프로시저에 대한 호출을 생성합니다. extend 인자와 함께 호출되면 XmNgainPrimaryCallback 프로시저에 대한 호출을 생성할 수 있습니다. 자세한 정보는 *Motif Programmer's Reference*의 콜백 설명을 참조하십시오.

#### right-word(extend)

XmNeditPolicy가 XmEDIT\_LOGICAL이고 아무런 인자 없이 호출되면 삽입점을 논리적인 다음 단어가 존재하는 경우 해당 단어의 논리적 시작 부분으로 이동시킵니다. 그렇지 않으면 기존 단어의 논리적 끝 부분으로 이동시킵니다. 삽입 커서가 줄의 논리적인 끝 부분이나 줄의 논리적인 마지막 단어에 있으면 해당 커서를 다음 줄의 논리적인 첫 번째 단어로 이동시킵니다. 그렇지 않으면 기존 단어의 논리적 끝 부분으로 이동시킵니다.

XmNeditPolicy가 XmEDIT\_VISUAL이고 인자가 없이 호출되면 삽입 커서를 줄 끝의 오른쪽 또는 뒤에 있는 첫 번째 흰색 공백 문자 뒤 비 흰색 공백 문자로 이동시킵니다.

extend의 인자를 통해 호출되면 인자가 없는 경우와 마찬가지로 삽입 커서를 이동시키고 현재의 선택을 확장합니다.

left-word() 활동은 합리적인 값 XmCR\_MOVING\_INSERT\_CURSOR을 가진 XmNmotionVerifyCallback 프로시저에 대한 호출을 생성합니다. extend 인자를 통해 호출되면 XmNgainPrimaryCallback 프로시저에 대한 호출을 생성할 수 있습니다. 자세한 정보는 *Motif Programmer's Reference*의 콜백 설명을 참조하십시오.

#### delete-left-character()

XmNeditPolicy가 XmEDIT\_LOGICAL이면 delete-previous-char()에 상응합니다. XmNeditPolicy가 XmEDIT\_VISUAL이면 정상 모드에서 널이 아닌 선택이 있는 경우 해당 선택을 삭제합니다. 그렇지 않으면 삽입 커서의 남은 문자를 삭제합니다. 추가 모드에서 널이 아닌 선택이 있으면 커서는 선택에서 사라지지 않고 XmNpendingDelete가 참으로 설정되며 해당 선택을 삭제합니다. 그렇지 않으면 삽입 커서의 남은 문자를 삭제합니다. 이것은 선택에 영향을 줄 수 있습니다.

delete-left-character() 활동은 합리적인 값 XmCR\_MODIFYING\_TEXT\_VALUE을 가진 XmNmodifyVerifyCallback 프로시저 및 합리적인 값 XmCR\_VALUE\_CHANGED을 가진 XmNvalueChangedCallback 프로시저에 대한 호출을 생성합니다.

#### delete-right-character()

XmNeditPolicy가 XmEDIT\_VISUAL이면 delete-next-character()에 상응합니다. XmNeditPolicy가 XmEDIT\_VISUAL이면 정상 모드에서 널이 아닌 선택이 있는 경우 해당 선택을 삭제합니다. 그렇지 않으면 삽입 커서의 오른쪽 문자를 삭제합니다. 추가 모드에서 널이 아닌 선택이 있으면 커서는 선택에서 사라지지 않고

XmNpendingDelete가 참으로 설정되며 해당 선택을 삭제합니다. 그렇지 않으면 삽입 커서의 오른쪽 문자를 삭제합니다. 이것은 선택에 영향을 줄 수 있습니다.

delete-left-character() 활동은 합리적인 값 XmCR\_MODIFYING\_TEXT\_VALUE을 가진 XmNmodifyVerify-Callback 프로시저 및 합리적인 값 XmCR\_VALUE\_CHANGED을 가진 XmNvalue-ChangedCallback 프로시저에 대한 호출을 생성합니다.

문자 구성, 묶음 기호 및 분음 부호를 지원하기 위해 일부 셀 기반 루틴이 구축됩니다. 다시 말해, 두 개 이상의 문자가 단일 표시 셀을 점유하는 도형 문자로 표현됩니다.

XmText 셀 활동 루틴은 다음과 같습니다:

backward-cell (extend)

삽입 커서를 한 셀 뒤로 이동합니다. XmNeditPolicy가 XmEDIT\_LOGICAL이면 삽입 커서는 현재의 셀보다 논리적으로 앞선 셀의 시작 부분으로 이동합니다. 그렇지 않으면 현재 셀의 시작 부분으로 이동합니다.

XmNeditPolicy가 XmEDIT\_VISUAL이면 커서는 커서 왼쪽 셀의 시작 부분으로 이동합니다. prev-cell() 활동은 합리적인 값 XmCR\_MOVING\_INSERT\_CURSOR을 가진 XmNmotionVerifyCallback 프로시저에 대한 호출을 생성합니다. extend 인자를 통해 호출되면 XmNgainPrimaryCallback 프로시저에 대한 호출을 생성할 수 있습니다. 자세한 정보는 *Motif Programmer's Reference*의 콜백 설명을 참조하십시오.

forward-cell (extend)

삽입 커서를 논리적인 다음 셀의 시작 부분으로 이동시킵니다. 그렇지 않으면 해당 셀의 끝으로 이동시킵니다. XmNeditPolicy가 XmEDIT\_LOGICAL이면 커서는 한 셀 앞으로 이동합니다.

XmNeditPolicy가 XmEDIT\_VISUAL이면 커서는 커서 위치 오른쪽 셀의 시작 부분으로 이동합니다. 그렇지 않으면 현재 셀의 끝 부분으로 이동합니다. forward-cell() 활동은 합리적인 값 XmCR\_MOVING\_INSERT\_CURSOR을 가진 XmNmotionVerifyCallback 프로시저에 대한 호출을 생성합니다. extend 인자를 통해 호출되면 XmNgainPrimaryCallback 프로시저에 대한 호출을 생성할 수 있습니다. 자세한 정보는 *Motif Programmer's Reference*의 콜백 설명을 참조하십시오.

---

## XmTextFieldGetLayoutModifier

XmTextFieldGetLayoutModifier()는 rendition에 묶여 있는 레이아웃 객체의 상태를 반영하는 레이아웃 수정자 문자열을 반환합니다.

XmTextFieldGetLayoutModifier()용 구문:

```
#include <Xm/TextF.h>
String XmTextFieldGetLayoutModifier ( Widget widget)
```

`XmTextFieldGetLayoutModifier()`는 해당 위젯과 관련된 `rendition`의 현재 레이아웃 객체 설정 값에 액세스합니다. 편리한 함수를 이용해 레이아웃 객체 수정자 값이 변경되면 `XmTextFieldGetLayoutModifier` 함수가 변경된 값 뿐 아니라 레이아웃 객체의 전체 상태를 반환합니다.

`XmTextFieldGetLayoutModifier()`는 문자열 값 형태로 레이아웃 객체 수정자 값을 반환합니다.

---

## XmTextGetLayoutModifier

`XmTextGetLayoutModifier()`는 `rendition`에 묶여 있는 레이아웃 객체의 상태를 반영하는 레이아웃 수정자 문자열을 반환합니다.

`XmTextGetLayoutModifier()`용 구문:

```
#include <Xm/Text.h>
String XmTextGetLayoutModifier( Widget widget )
```

`XmTextGetLayoutModifier`는 해당 위젯과 관련된 `rendition`의 현재 레이아웃 객체 설정 값에 액세스합니다. 편리한 함수를 사용해 레이아웃 객체 수정자 값을 변경하면 `XmTextGetLayoutModifier` 함수는 변경된 값 뿐 아니라 전체 레이아웃 객체의 상태를 반환합니다.

`XmTextGetLayoutModifier`는 문자열 값의 형태로 레이아웃 객체 수정자 값을 반환합니다.

---

## XmTextFieldSetLayoutModifier

`XmTextFieldSetLayoutModifier()`는 `rendition`에 묶여 있는 레이아웃 객체의 동작을 변경하는 레이아웃 수정자 값을 설정합니다.

`XmTextFieldSetLayoutModifier()`용 구문:

```
#include <Xm/TextF.h>
void XmTextFieldSetLayoutModifier( widget widget , string layout_modifier)
```

`XmTextFieldSetLayoutModifier`는 위젯과 관련된 `rendition`의 레이아웃 객체 설정을 수정합니다. 편리한 함수를 사용해 레이아웃 객체 수정자 값이 설정되면 입력 매개 변수에 지정된 속성들만이 변경됩니다. 그 밖의 속성은 바뀌지 않습니다.

---

## XmTextSetLayoutModifier

`XmTextSetLayoutModifier()`는 rendition에 묶여 있는 레이아웃 객체의 동작을 변경하는 레이아웃 수정자 값을 설정합니다.

`XmTextSetLayoutModifier()` 용 구문:

```
#include <Xm/Text.h>
void XmTextSetLayoutModifier( Widget widget , string layout_modifier)
```

`XmTextSetLayoutModifier`는 위젯과 관련된 rendition의 레이아웃 객체 설정을 수정합니다. 편리한 함수를 사용해 레이아웃 객체 수정자 값이 설정되면 입력 매개변수에 지정된 속성들만이 변경됩니다;그 밖의 속성은 바뀌지 않습니다.

---

## XmStringDirectionCreate

`XmStringDirectionCreate`는 복합 문자열을 만듭니다.

`XmTextSetLayoutModifier()` 용 구문:

```
#include <Xm/Xm.h>
XmString XmStringDirectionCreate( direction)
XmStringDirection direction
```

`XmStringDirectionCreate`는 단일 구성 요소인 주어진 값의 방향이 있는 복합 문자열을 만듭니다. 한편, `XmNlayoutDirection` 자원은 문자열을 위한 방향을 지정하는 구성 요소를 갖고 있지 않은 임의의 복합 문자열(`XmString`)을 위한 기본 렌더링 방향을 설정합니다. 따라서 레이아웃 방향을 설정하려는 경우 해야 할 일은

`XmNlayoutDirection` 자원의 적절한 값을 설정하는 것뿐입니다. 특정 방향 구성 요소를 갖는 복합 문자열을 만들 필요는 없습니다. 응용프로그램이 `XmString`을 렌더링할 때는 해당 문자열이 명시적인 방향(`XmStringDirection`)을 갖도록 작성되었는지 확인하게 됩니다. 아무런 방향 구성 요소가 없다면 응용프로그램은 현재 위젯용 `XmNlayoutDirection` 자원의 값을 확인하여 해당 값을 `XmString` 용 기본 렌더링 방향으로 사용합니다.

---

## UIL Arguments

다음 표는 UIL 인자를 보여줍니다.

표 6-3 UIL

UIL 인자 이름	인자 유형
XmNlayoutAttrObject	문자열
XmNlayoutModifier	문자열
XmNrenditionTag	문자열
XmNalignment	정수
XmNeditPolicy	정수

---

---

## CTL 응용프로그램 개발 방법

다음 절에서는 CTL 응용프로그램 개발법을 설명합니다.

### 레이아웃 방향

복합 문자열의 방향은 영어, 스페인어, 프랑스어 및 독일어와 같이 왼쪽에서 오른쪽으로 쓰는 언어나 히브리어 및 아랍어와 같은 오른쪽에서 왼쪽으로 쓰는 언어의 텍스트 모두에 데이터 구조가 유용하도록 저장됩니다. Motif 응용프로그램에서는 VendorShell 또는 MenuShell에서 XmNlayoutDirection 자원을 사용해 레이아웃 방향을 설정할 수 있습니다. 관리자 및 프리미티브 위젯(Gadgets 포함) 역시 XmNlayoutDirection 자원을 갖고 있습니다. 기본 값은 동일한 자원을 갖고 있는 가장 가까운 상위로부터 상속됩니다.

XmText 위젯의 경우 세로 방향도 지정해야 합니다. layoutDirection을 XmRIGHT\_TO\_LEFT로 설정하면 문자열 방향은 오른쪽에서 왼쪽으로 설정되지만 커서는 세로 방향으로 움직입니다. 세로 방향이 중요하고 위에서 아래로 정렬할 필요가 있으면 반드시 XmRIGHT\_TO\_LEFT TOP\_TO\_BOTTOM 을 지정하십시오. 이 경우 구성 요소들은 오른쪽에서 왼쪽으로, 그리고 위에서 아래로 배치되어 원하는 동작이 이루어집니다.

아울러 XmText 및 TextField 위젯의 동작은 XmRendition의 XmNalignment 및 XmNlayoutModifier 자원의 영향을 받습니다. 이 자원들은 XmNlayoutDirection과 함께 텍스트 위젯의 레이아웃 동작을 제어합니다. 이 동작은 그림 6-2에 표시되어 있습니다.

그림에서 사용된 입력 문자열:

A B و ض

다음 그림을 위한 XmNlayoutModifier 문자열 @ls orientation= 설정 값은 왼쪽 열에 표시되어 있습니다.

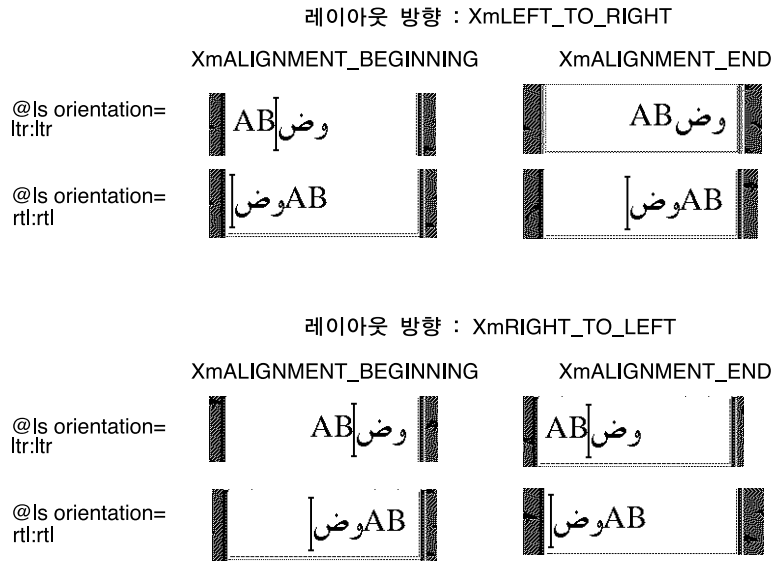


그림 6-2 레이아웃 방향

그림에서 보이는 바와 같이, XmNAlignment는 레이아웃 방향에 따라 텍스트 오른쪽을 비울 것인지, 아니면 왼쪽을 비울 것인지를 지정합니다. XmNlayoutModifier는 텍스트를 세그먼트로 나눈 뒤 방향 값에 따라 왼쪽에서 오른쪽 또는 오른쪽에서 왼쪽으로 배열합니다. 다시 말해 XmNlayoutDirection이 XmRIGHT\_TO\_LEFT 이고 XmNAlignment 값이 XmALIGNMENT\_BEGINNING 이면 문자열 오른쪽이 비워집니다.

---

## Rendition 만들기

다음 코드는 XmNlabelString이 유형 XmCHARSET\_TEXT 인 XmLabel을 태그가 "ArabicShaped."인 Rendition을 사용해 만듭니다. Rendition은 "ar" (아랍어 로케일용 로케일 이름에 해당)의 XmNlayoutAttrObject 와 출력 버퍼용으로 NUMERALS\_CONTEXTUAL의 Numerals 값과 "유니코드-3.0."의 ShapeCharset 값을 지정하는 레이아웃 수정자를 사용해 작성됩니다.

로케일 지향 레이아웃 모듈은 입력 텍스트를 16비트 유니코드 3.0 코드 세트를 사용해 코드화된 실제 문자의 출력 버퍼로 전송합니다. 명시적 레이아웃 로케일은 지정되어 있기 때문에 이 텍스트는 런타임 로케일 설정에 관계 없이 올바르게 렌더링됩니다. 이 예에서는 입력이 ISO 8859-6으로 코드화됩니다.

```
int n;
Arg args[10];
Widget w;
XmString labelString;
XmRendition rendition;
XmStringTag renditionTag;
XmRenderTable renderTable;
    /* alef lam baa noon taa - iso8859-6 */
labelString = XmStringGenerate("\307\344\310\346\312", NULL
                                XmCHARSET_TEXT, "ArabicShaped");
w = XtVaCreateManagedWidget("a label", xmLabelWidgetClass, parent,
                            XmNlabelString, labelString,
                            XmNlabelType, XmSTRING,
                            NULL);
n = 0;
XtSetArg(args[n], XmNfontName, "-*-medium-r-normal-*-24-*-*-*-*");
n++;
XtSetArg(args[n], XmNfontType, XmFONT_IS_XOC); n++;
XtSetArg(args[n], XmNlayoutAttrObject, "ar"); n++;
XtSetArg(args[n], XmNlayoutModifier,
         "@ls numerals=:contextual, shapecharset=iso8859-6"); n++;
renditionTag = (XmStringTag) "ArabicShaped";
rendition = XmRenditionCreate(w, renditionTag, args,
s, n);
renderTable =
    XmRenderTableAddRenditions(NULL, &rendition, 1, XmREPLACE_MERGE);
XtVaSetValues(w, XmNrenderTable, renderTable, NULL);
```

## Rendition 편집하기

다음 코드는 TextField 위젯과 RenderTable을 단일 Rendition으로 만듭니다. XmNlayoutAttrObject 및 XmNlayoutModifier 의사 자원은 지정되지 않은 채로 남았으므로 NULL로 설정됩니다. 이 값은 Rendition 명령과 관련된 레이아웃 객체가 존재하는 경우 기본 로케일에 속한다는 것을 의미합니다.

이 예제가 올바르게 작동하려면 로캘을 코드 세트가 ISO 8859-6이고 로캘 지향 레이아웃 모듈이 IMPLICIT\_BASIC 알고리즘을 지원할 수 있는 로캘로 설정되어야 합니다. Rendition의 LayoutObject의 ImplicitAlg 값은 Rendition의 XmNlayoutModifier 의사 자원에 의해 수정됩니다.

```
int n;
Arg args[10];
Widget w;
    XmRendition rendition;
XmStringTag renditionTag;
XmRenderTable renderTable;
w = XmCreateTextField(parent, "text field", args, 0);
n = 0;
    XtSetArg(args[n], XmNfontName, "-*-medium-r-normal-*-24-*-*-*-*-*");
    n++;
    XtSetArg(args[n], XmNfontType, XmFONT_IS_XOC); n++;
renditionTag = (XmStringTag) "ArabicShaped";
rendition = XmRenditionCreate(w, renditionTag, args, n);
renderTable =
    XmRenderTableAddRenditions(NULL, &rendition, 1, XmREPLACE_MERGE);
XtVaSetValues(w, XmNrenderTable, renderTable, NULL);
    ....
n = 0;
XtSetArg(args[n], XmNlayoutModifier, "@ls implicitalg=basic");
    n++;
XmRenditionUpdate(rendition, args, n);
```

---

## 리소스 파일에서 렌더 테이블 만들기

응용프로그램의 국제화가 올바르게 이루어지려면 Rendition과 렌더 테이블을 지정해야 합니다. 렌더 테이블이 파일에서 지정되면 프로그램 바이너리는 주어진 로캘의 요건에 독립적으로 만들어지므로 로캘 요건에 맞추어 쉽게 사용자 정의할 수 있습니다.

렌더 테이블은 다음 구문에 의해 리소스 파일에서 지정됩니다:

```
resource_spec: [tag [, tag] *]
```

이 때 *tag*는 rendition의 XmNtag 자원에 적합한 일부 문자열입니다.

이 줄은 지정된 바와 같이 한 개 이상의 rendition을 포함한 최초의 렌더 테이블을 만듭니다. rendition은 지정된 태그에 부착됩니다:

```
resource_spec [* | .] rendition [* | .] resource_name: value
```

다음 예제에서는 리소스 파일을 사용해 설정할 수 있는 XmRendition과 관련된 CTL 자원들을 보여줍니다. 레이아웃 객체가 효과를 발휘하려면 fontType을 FONT\_IS\_XOC로 설정해야 합니다. @ls를 사용해 지정된 layoutModifier는 rendition 객체에 의해 레이아웃 객체에 전달됩니다.



layoutModifier를 이용해 레이아웃 객체에 설정될 수 있는 리소스의 전체 목록을 보려면, *CAE Specification: Portable Layout Services: Context-dependent and Directional Text* (Open Group: Feb 1997; ISBN 1-85912-142-X; 문서 번호 C616)를 참조하십시오.

---

## 응용프로그램에서 렌더 테이블 만들기

렌더 테이블을 만들기 전에 먼저 응용프로그램이 테이블의 일부인 rendition 중 최소 한 개를 작성해야 합니다. XmRenderTableAddRenditions() 기능은 이름이 암시하는 바와 같이 새 rendition으로 렌더 테이블을 증대시키는 것입니다. 새 렌더 테이블을 만들려면 기존 렌더 테이블 대신 NULL 인자로 XmRenderTableAddRenditions() 기능을 호출하십시오.

다음 코드는 XmFONT\_IS\_XOC에 설정된 XmNfontType으로 만든 rendition을 사용해 렌더 테이블을 만듭니다.

```
int n;
Arg args[10];
Widget w;
XmString labelString;
XmRendition rendition;
XmStringTag renditionTag;
XmRenderTable renderTable;
/* alef lam baa noon taa - iso8859-6 */
labelString = XmStringGenerate("\307\344\310\346\312\312", NULL,
                               XmCHARSET_TEXT, "ArabicShaped");
w = XtVaCreateManagedWidget("a label", xmLabelWidgetClass, parent,
                             XmNlabelString, labelString,
                             XmNlabelType, XmSTRING,
                             NULL);
n = 0;
XtSetArg(args[n], XmNfontName, "-*-medium-r-normal-*-24-*-*-*-*-*");
n++;
XtSetArg(args[n], XmNfontType, XmFONT_IS_XOC); n++;
XtSetArg(args[n], XmNlayoutAttrObject, "ar"); n++;
XtSetArg(args[n], XmNlayoutModifier,
          "@ls numerals=nominal:contextual, shapecharset=iso8859-6"); n++;
renditionTag = (XmStringTag) "ArabicShaped";
rendition = XmRenditionCreate(w, renditionTag, args, n);
renderTable =
    XmRenderTableAddRenditions(NULL, &rendition, 1, XmREPLACE);
XtVaSetValues(w, XmNrenderTable, renderTable, NULL);
```

---

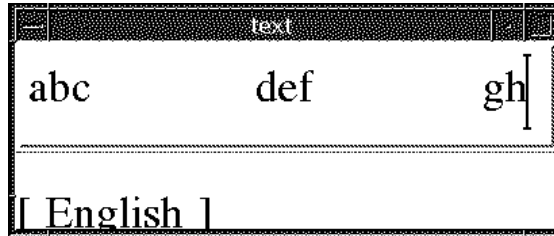
## 수평 탭

텍스트 배치를 제어하기 위해 복합 문자열에는 탭 문자가 포함됩니다. 화면의 문자들을 해석하기 위해 위젯트는 해당 복합 문자열을 위해 유효한 rendition을 참조하며 이 때 탭 스톱 목록을 찾게 됩니다. 하지만 동적 위젯트인 TextField 및 XmText는 rendition의 탭 자원을 사용하지 않습니다. 대신  $8 * (\text{width of character } 0)$  공식을 사용해 탭 너비를 계산합니다.

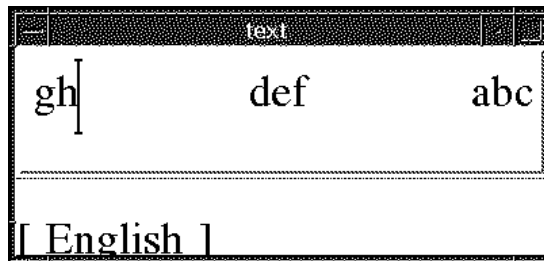
탭 측정치는 복합 문자열 표시의 왼쪽 여백으로부터의 거리입니다. 레이아웃 방향이 오른쪽에서 왼쪽으로인 경우 거리는 오른쪽 여백으로부터 측정됩니다. 텍스트의 방향에 관계 없이(아랍어 오른쪽에서 왼쪽, 또는 영어 왼쪽에서 오른쪽), 탭은 레이아웃 방향(XmNLayoutDirection)에 의해 지정된 데로 오른쪽이나 왼쪽에 공백을 삽입합니다.

탭 뒤의 텍스트는 항상 탭 스톱에 맞추어집니다. 탭 스톱은 위젯트의 시작 위치부터 계산되며 XmNLayoutDirection의 영향을 받습니다. 탭의 동작과 텍스트 방향성 및 위젯트의 XmNLayoutDirection과의 상호 작용은 다음 그림에 나타나 있습니다.

이 그림을 위한 입력은 abc\tdef\tgh입니다.



레이아웃 방향 : XmLEFT\_TO\_RIGHT



레이아웃 방향 : XmLEFT\_TO\_RIGHT

그림 6-3 탭 이동 동작

## 마우스 선택

사용자가 마우스 버튼 1을 사용해 1차 선택을 합니다. 버튼을 누르면 기존 선택은 취소되고 삽입 커서와 앵커가 버튼을 누른 텍스트 안의 위치로 이동합니다. 마우스 버튼 1을 누른 채 끌면 앵커와 포인터 위치 사이의 모든 텍스트가 선택되고 해당 범위 밖의 텍스트는 선택 해제됩니다.

선택된 텍스트는 XmNeditPolicy 자원의 영향을 받는데 이 자원은 XmEDIT\_LOGICAL 또는 XmEDIT\_VISUAL로 설정할 수 있습니다. 만일 XmNeditPolicy가 XmEDIT\_LOGICAL로 설정되어 있고 선택된 텍스트가 양방향이면 선택된 텍스트는 연속적이지 않은 세그먼트의 모음이 됩니다. 이것은 논리 버퍼의 텍스트가 디스플레이와 1 대 1 대응이 이루어지지 않기 때문입니다.

결과적으로 양방향 텍스트의 논리적 문자의 연속적 버퍼는 렌더링될 때 연속적인 문자 스트림으로 만들어지지 않습니다. 거꾸로, XmNeditPolicy가 XmEDIT\_VISUAL로 설정되면 선택된 텍스트는 시각적으로 연속적일 수 있지만 논리적 버퍼에서 세그먼트화되지 않습니다. 따라서 동일한 커서 포인트에서의 양방향 텍스트의 선택, 삭제 및 삽입 순서는 항상 동일한 결과를 가져오지는 않습니다.

---

## 키보드 선택

선택 작업은 마우스와 키보드로 수행할 수 있습니다. Shift 키와 화살표 키를 조합해서 텍스트를 선택할 수 있습니다.

선택된 텍스트는 자원의 영향을 받으며 `XmNeditPolicy` 자원은 `XmEDIT_LOGICAL` 또는 `XmEDIT_VISUAL`로 설정할 수 있습니다. `XmNeditPolicy`가 `XmEDIT_LOGICAL`로 설정되고 선택된 텍스트가 양방향이면 선택된 텍스트는 시각적으로 연속적인 것으로 보이지 않습니다. 논리 버퍼의 텍스트가 디스플레이와 1대1 대응이 이루어지지 않기 때문에 양방향 텍스트의 논리적 문자의 연속적 버퍼는 렌더링될 때 연속적인 문자 스트림을 만들어내지 않습니다.

거꾸로, `XmNeditPolicy`가 `XmEDIT_VISUAL`로 설정되면 선택된 텍스트는 시각적으로 연속적일 수 있지만 논리적 버퍼에서 세그먼트화되지 않습니다. 따라서 동일한 커서 포인트에서의 양방향 텍스트의 선택, 삭제 및 삽입 순서는 항상 동일한 결과를 가져오지는 않습니다.

---

## 텍스트 자원 및 기하학

기하학과 관련된 텍스트 자원은 다음과 같습니다:

- 렌더 테이블 `XmNrenderTable`은 글꼴이나 글꼴 집합 및 텍스트를 표시하기 위한 기타 속성을 선택하기 위해 위젯가 사용합니다.

`Text` 및 `Textfield` 위젯트는 `XmNfontType` 과 같이 글꼴과 관련된 표현 자원만을 사용합니다. 이 위젯트는 보통 로케일 식별자인 `XmNlayoutAttrObject`와 이 `XmNrendition`과 관련된 `XOC`와 관련된 레이아웃 객체에 전달할 레이아웃 값을 지정하는 `XmNlayoutModifier`같은 레이아웃 객체의 속성도 지정할 수 있습니다.

- 텍스트가 위젯트보다 넓은 경우 단어 경계에서 줄을 바꿀지를 지정하는 자원 (`XmNwordWrap`).

단어 경계에서 행을 바꾸어도 텍스트에 새 행이 삽입되지 않습니다. 아랍어와 같은 초서체 언어의 경우, 단어 길이가 위젯트 길이보다 길면 단어는 다음 줄로 줄바꿈되지만 다음 줄의 첫 번째 문자는 논리 버퍼의 이전 문자와 독립적인 모양을 갖게 됩니다.

---

## 이식 관련 지침

Complex Text Layout (CTL)용으로 활성화된 새 Motif 라이브러리는 `/usr/dt/lib/libXm.so.4`에 있습니다. 응용프로그램이 `libXm.so.3`에 연결되면 CTL을 지원하지 않습니다. `ldd app_name`은 응용프로그램이 연결할 라이브러리를 보여줍니다. CTL을 활성화하기 위해 기존 응용프로그램을 이식하려면 다음 단계를 따라야 합니다.

1. `-DSUN_CTL`을 `Makefile`에 추가하십시오. 이 플래그는 중요한 것으로 CTL을 지원하기 위해 필요한 데이터 구조를 포함하고 있습니다. 이것은 컴파일 과정에서 설정해야 합니다.
2. 기존 응용프로그램을 다시 컴파일하십시오. 컴파일을 다시 하면 자동으로 CTL을 사용할 수 있는 Motif 라이브러리 `libXm.so.4`에 연결됩니다.
3. `XmText.translations` 자원을 응용프로그램 리소스 파일에 추가하십시오. 리소스가 없으면 로케일의 레이아웃 엔진이 실행되지 않습니다.
4. 설명서에 첨부된 예제 응용프로그램을 참조하십시오.

---

주 - `fontName` 자원에서 자신의 로케일에 적합한 사용 가능한 글꼴 이름을 사용하십시오.

---

예를 들어, `XmTextField` 또는 `XmText` 위젯트에서 셀 기반 문자 이동(태국어)을 원하면 해당 위젯트의 번역을 다음과 같이 설정하십시오:

```
XmText.translations: #override \n\  
<Key>osfRight:forward-cell() \n\  
<Key>osfLeft:backward-cell() \n\  
<Key>osfDelete:delete-next-cell() \n\  
<Key>osfBackSpace:delete-previous-cell() \n\  

```



# mp를 사용한 인쇄 필터 강화

---

이 장은 mp를 사용한 인쇄 강화에 대해 설명합니다. 다음 항목을 다룹니다.

- “UTF-8에 대한 인쇄” 143 페이지
- “mp 인쇄 필터 강화 개요” 144 페이지
- “구성 파일의 지역화” 145 페이지
- “로케일 종속 프롤로그 파일” 152 페이지
- “기존 프롤로그 파일 사용자 정의 및 새 프롤로그 파일 추가하기” 151 페이지
- “포스트스크립트 파일 사용자 정의” 151 페이지
- “.xpr File Customization” 154 페이지
- “새 .xpr 파일 작성하기” 157 페이지

---

## UTF-8에 대한 인쇄

Solaris 9 운영 환경에서, UTF-8으로 작성된 텍스트 파일을 포함한 다양한 입력 형식의 파일을 인쇄할 수 있게 해주는 mp(1) 인쇄 필터가 더욱 개선되었습니다. 이 필터는 Solaris 운영 체제에서 사용할 수 있는 트루타입과 유형 1 조정 가능 글꼴 및 X11 비트맵 글꼴을 사용합니다. 또한 프린터 상주 글꼴을 사용할 수도 있고 X 인쇄 서버 클라이언트 역할을 할 수 있습니다.

유틸리티를 사용하여 표준 포스트스크립트로 출력한 뒤 포스트스크립트 프린터로 전송할 수 있습니다. mp(1)은 X 인쇄 서버 클라이언트로 작동할 때 모든 페이지 설명 언어를 출력할 수도 있고 mp 는 인쇄 서버의 지원을 받습니다.

유틸리티를 사용하려면 다음을 입력하십시오.

```
system% mp filename | lp
```

또한 이 유틸리티는 stdin 스트림을 받아들이기 때문에 필터로 사용할 수도 있습니다.

```
system% cat filename | mp | lp
:
```

이 유틸리티를 라인 프린터용 인쇄 필터로 설정할 수 있습니다. 예를 들어, 다음 명령 시퀀스는 프린터 서비스 LP에 프린터 lp1이 mp 형식의 파일만을 받아들인다고 알려줍니다. 이 명령은 또한 프린터 lp1을 /dev/ttya 포트에 설치합니다. 자세한 내용은 lpadmin(1M) 설명서 페이지를 참조하십시오.

```
system# lpadmin -p lp1 -v /dev/ttya -I MP
system# accept lp1
system# enable lp1
```

lpfilter(1M)을 사용하여 필터에 대한 유틸리티를 다음과 같이 추가할 수 있습니다.

```
system# lpfilter -f lp1 -F pathname
```

이 명령은 필터 설명 파일인 *pathname*을 통해 변환기(이 경우에는 mp)를 사용할 수 있음을 LP에 알려줍니다. *pathname*은 다음을 포함합니다.

```
Input types: simple
Output types: MP
Command: /usr/bin/mp
```

필터는 /usr/bin/mp를 사용하여 기본 유형 파일 입력을 포스트스크립트 출력으로 변환합니다.

UTF-8 텍스트 파일을 인쇄하려면 다음 명령을 사용하십시오.

```
system% lp -T MP UTF-8-file
```

자세한 내용은 mp 설명서 페이지를 참조하십시오.

---

## mp 인쇄 필터 강화 개요

mp 인쇄 필터가 Solaris 9 릴리스에서 강화됩니다. 최신 mp 는 로켈의 출력 파일을 만드는 세 개의 다른 모드에서 내부적으로 작동하여 국제 텍스트를 인쇄합니다. 사용 가능한 모드는 다음과 같습니다.

- 로켈 특정 글꼴 구성 파일 mp.conf와 작동하는 mp.
- 로켈 특정 포스트스크립트 프롤로그 파일 prolog.ps와 작동하는 mp.
- Xprt(X 인쇄 서버) 클라이언트로 작동하는 mp.

다음 절은 특정 인쇄 방법을 사용하는 시기와 이러한 인쇄 방법에 대하여 mp가 어떤 구성 및 지원 파일을 사용하는지 설명합니다.

## 로켈 특정 구성 파일 mp.conf와 함께 mp 사용하기

만약 -D나 -p가 명령줄에서 주어지지 않았고, prolog.ps 파일이 /usr/openwin/lib/locale/\$LANG/print 또는 /usr/lib/lp/locale/\$LANG/mp에도 없다면 로켈 특정 글꼴 구성 파일인



mp.conf를 기본적으로 사용합니다. prolog.ps 파일은 파일의 포스트스크립트 내장 글꼴을 사용하여 mp를 인쇄하게 합니다. prolog.ps 가 로캘이 존재하더라도 -M 옵션을 사용하면 prolog.ps 파일이 무시되고 대신 mp.conf 파일이 사용됩니다.

이 방법은 /usr/lib/lp/locale/\$LANG/mp/mp.conf 글꼴 구성 파일을 사용합니다. 대체 글꼴을 사용해야 하는 경우를 제외하고 이 파일을 변경하지 않아도 됩니다. 이 파일은 트루타입, 유형 1 또는 pcf 글꼴로 구성될 수 있습니다. /usr/lib/lp/locale/C/는 다음 메소드는 물론 이 인쇄 방법에 공통적인 .ps 인쇄 페이지 레이아웃 파일을 포함합니다. 파일을 사용자 정의하는 방법에 대한 설명은 “기존 프롤로그 파일 사용자 정의 및 새 프롤로그 파일 추가하기” 151 페이지에서 제공됩니다.

## 로캘 특정 포스트스크립트 프롤로그 파일과 함께 mp 사용하기

만약 -D 또는 -P가 명령줄에서 주어지지 않았고, /usr/openwin/lib/locale/\$LANG/print/prolog.ps가 존재하면 prolog.ps 파일이 출력을 위해 미리 결정됩니다. .ps 프롤로그 페이지의 인쇄 스타일에 따라 레이아웃 파일도 출력을 위해 미리 결정됩니다.

이 인쇄 방법은 포스트스크립트 글꼴 파일만 사용합니다. prolog.ps 파일의 사용자 정의는 “Xprt(X 인쇄 서버) 클라이언트로 mp 사용하기” 145 페이지에서 설명됩니다.

## Xprt(X 인쇄 서버) 클라이언트로 mp 사용하기

이 지원은 mp가 X 인쇄 서버가 지원하는 네트워크에 연결된 모든 프린터에 대한 출력을 인쇄할 수 있도록 합니다. 포스트스크립트와 많은 PLC 버전도 이 명령으로 지원됩니다.

-D 또는 -P 명령 옵션이 사용되고 XPDISPLAY 변수가 환경에 설정되어 있지 않으면 인쇄 서버 시작 스크립트가 클라이언트가 실행 중인 컴퓨터에서 포트 2100로 Xprt 서버를 시작합니다. 이 스크립트는 또한 mp가 완료된 후 인쇄 서버를 종료합니다. XPDISPLAY가 설정되면 mp 클라이언트는 XPDISPLAY에서 실행되는 인쇄 서버에 접속하려 합니다. 이 경우 서버가 실행 중이 아니면 서버를 시작하려는 시도가 이루어지지 않습니다.

/usr/lib/lp/locale/C/mp 디렉토리는 Xprt 클라이언트로 작동하는 mp에 대한 .xpr 인쇄 페이지 레이아웃 파일을 포함합니다. 이들은 300dpi 프린터를 위해 작성된 예제 파일입니다. 대상 프린터가 다른 dpi 값을 가지면 dpi 값은 자동으로 대상 프린터의 해상도로 변환됩니다.

## 구성 파일의 지역화

구성 파일은 글꼴 항목 또는 글꼴 그룹 항목을 추가 또는 변경하는 유연성을 제공합니다.

시스템 기본 구성 파일 `/usr/lib/lp/locale/$LANG/mp/mp.conf`, 여기서 `$LANG`은 인쇄가 이루어지는 로케일의 로케일 환경 변수입니다. 사용자는 `-u config:file path` 옵션으로 개인 구성 파일을 지정하여 사용할 수 있습니다.

호환용 문자로 인코딩된 슬러 또는 변수 그림 문자를 **표현 형식**이라고 합니다. `mp.conf` 파일은 로케일의 코드 포인트를 그 코드 포인트를 인쇄하는데 사용되는 글꼴의 인코딩된 표현 형식으로 매핑하는데 주로 사용됩니다.

중간 코드 포인트는 넓은 문자이거나 이식 가능 레이아웃 서비스(PLS) 레이어의 출력일 수 있습니다. 복잡한 텍스트 레이아웃 인쇄를 위해서는 중간 코드 포인트가 PLS 출력이어야 합니다. `mp(1)`이 생성한 기본 중간 코드는 PLS 출력입니다.

현재 지원되는 글꼴 형식은 PCF(Portable Compiled Format), TrueType 및 Type1 형식입니다. 시스템 상주 및 프린터 상주 Type1 글꼴이 모두 지원됩니다. `mp.conf` 구성 파일의 형식과 내용에 대하여 다음을 기억하십시오.

- 행은 유효한 키보드(지시어)로 시작해야 합니다.
- 키보드에 대한 인자는 키워드와 동일한 행에 나타나야 합니다.
- # 문자로 시작되는 행은 행 끝까지 주석으로 취급됩니다.
- 0x로 시작되는 숫자 인자는 16진수 숫자로 해석됩니다.

`mp.conf` 파일의 다른 부분은 다음을 포함합니다.

- 글꼴 별명 짓기
- 글꼴 그룹 정의
- 로케일에서 중간 코드 범위에서 글꼴 그룹에 이르는 매핑
- 중간 코드 포인트를 글꼴 인코딩의 표현 형식으로 매핑하는 공유 객체와 각 글꼴을 연관

## 글꼴 별명 짓기

`mp.conf` 파일의 글꼴 별명 짓기 부분은 인쇄에 사용되는 각 글꼴의 별명을 정의하는데 사용됩니다. 이 절의 각 행은 다음과 같은 형식입니다.

`FontNameAlias` 글꼴-별명-이름 글꼴-유형 글꼴-경로

### 글꼴-별명-이름

글꼴 이름 별명 짓기의 일반적인 규약은 글꼴의 인코딩/스크립트 Roman, Bold, Italic 또는 BoldItalic (R, B, I 또는 BI)인지를 나타내는 문자가 뒤에 오도록 지정하는 것입니다.

예를 들어, `/usr/openwin/lib/X11/fonts/75dpi/courR18.pcf.Z`, 이것은 iso88591 로마 글꼴이기 때문에 별명 `iso88591R`이 주어질 수 있습니다.

### 글꼴-유형

.pcf 글꼴에 대하여 PCF, Adobe Type1 글꼴에 대하여 Type1, TrueType 글꼴에 대하여 TrueType을 지정합니다. 이러한 세 가지 글꼴 종류만 이 `mp.config` 파일에서 구성될 수 있습니다.

### 글꼴-경로

글꼴 파일에 절대 경로 이름을 부여합니다. Type1 프린터 상주 글꼴에 대해서는 Helvetica와 같은 글꼴 이름을 지정합니다.

예를 들어,

```
FontNameAlias prnHelveticaR Type1 Helvetica
```

### 글꼴 그룹 정의

동일한 유형의 글꼴을 조합하여 글꼴 그룹을 형성할 수 있습니다. 글꼴 그룹의 형식은 다음과 같습니다.

<b>키워드</b>	FontGroup
<i>fontgroupname</i>	글꼴의 그룹 이름
<i>GroupType</i>	글꼴 유형동일한 유형의 글꼴(PCF, Type1, TrueType)에 대해서만 글꼴 그룹을 만드십시오.
<i>Roman</i>	글꼴 그룹의 로마 글꼴 이름
<i>Bold</i>	글꼴 그룹의 굵은체 글꼴 이름
<i>Italic</i>	글꼴 그룹의 기울임꼴 글꼴 이름
<i>BoldItalic</i>	글꼴 그룹의 굵은 기울임 글꼴 이름.

그룹을 만들려면 로마 글꼴 항목만 필요합니다. **Bold**, **Italic** 및 **BoldItalic** 글꼴은 선택적입니다. 다른 유형의 글꼴은 예를 들어, 우편/뉴스 기사의 헤더 행을 표시하는데 사용됩니다. 로마 글꼴만 정의되면 다른 글꼴 대신 사용됩니다.

### 매핑 부분

mp.conf 파일의 매핑 부분은 로캘에서 중간 코드 범위를 글꼴 그룹에 매핑합니다. 이 부분의 각 행은 다음과 같은 형식입니다.

<b>키워드</b>	MapCode2글꼴.
<i>range_start</i>	0x로 시작되는 4바이트 16진수 값. 하나 이상의 글꼴 그룹에 매핑되는 코드 범위의 시작을 나타냅니다.
<i>range_end</i>	매핑될 코드 범위의 끝을 나타냅니다. 단일 중간 코드 포인트가 대상 글꼴에 매핑될 때는 '-' 값만 가능합니다.
<b>그룹</b>	표현 형식이 인쇄될 Type1, PCF 또는 TrueType 글꼴 그룹.

## 연관 부분

mp.conf 파일의 연관 부분은 각 글꼴을 글꼴 인코딩의 표현 형식에 중간 코드 포인트를 매핑하는 공유 객체와 연관시킵니다. 이 부분의 각 행은 다음과 같은 형식입니다.

키워드	CnvCode2Font.
글꼴 별명	글꼴에 정의된 별명
매핑 함수	중간 코드를 받아들이고 글꼴 인코딩의 표현 형식을 반환하면 이것은 그림 문자 색인을 얻고 그림 문자를 그리는 데 사용됩니다.
파일 경로 매핑 함수	매핑 기능을 포함하는 .so 파일 이름 dumpcs를 사용하여 EUC 로캘의 중간 코드 세트를 찾을 수 있습니다.

---

주 - mp(1)이 사용하는 현재 트루타입 엔진은 형식 4와 PlatformID 3 cmap만 다룰 수 있습니다. 즉, Microsoft .ttf 파일만 구성할 수 있습니다. 추가로, 문자 맵 인코딩인 트루타입 글꼴 엔진이 제대로 작동하기 위해 유니코드 또는 기호여야 합니다. Solaris 환경의 .ttf 글꼴 대부분은 이러한 제약을 준수하기 때문에 mp.conf 파일 내에서 Solaris 소프트웨어의 모든 트루타입 글꼴을 매핑할 수 있습니다.

---

PCF type1 X 논리적 글꼴 설명(XLFD)에 해당하는 글꼴 매핑을 위해 공유 객체를 만들면 중간 코드 범위를 XLFD가 지정하는 인코딩으로 매핑하는 공유 객체를 만드십시오. 예를 들어, 다음과 같습니다.

```
-monotype-arial-bold-r-normal-bitmap-10-100-75-75-p-54-iso8859-8
```

해당 PCF 글꼴은 다음과 같습니다.

```
/usr/openwin/lib/locale/iso_8859_8/X11/fonts/75dpi/ariabd10.pcf.Z
```

이 글꼴은 isoISO 8859-8로 인코딩되므로 공유 객체는 중간 코드와 해당 ISO 8859-8 코드 포인트 사이를 매핑해야 합니다.

XLFD가 있는 트루타입 글꼴의 경우

```
-monotype-arial-medium-r-normal--0-0-0-0-p-0-iso8859-8
```

해당 글꼴은 다음과 같습니다.

```
/usr/openwin/lib/locale/iso_8859_8/X11/fonts/TrueType/arial_h.ttf
```

이전 트루타입 글꼴의 cmap 인코딩이 유니코드로 되어 있기 때문에 중간 코드와 유니코드 사이를 매핑해야 합니다. 트루타입 글꼴의 예제에서 헤브루 문자(PLS 레이어가 만들)에 해당하는 en\_US.UTF-8 로캘 예제 중간 코드가 0xe50000e9라고 가정합니다. 글꼴이 유니코드로 인코딩되기 때문에 0xe50000e9를 전달할 때 출력이 유니코드의 표현 형식에 해당하는 방식으로 해당 .so 모듈 내에서 함수를 설계하십시오. 현재 예제는 0x000005d9입니다.

매핑 함수에 대한 함수 프로토타입은 다음과 같아야 합니다.

```
unsigned int function(unsigned int inter_code_pt)
```

다음은 mp.conf에서 사용할 수 있는 선택적 키워드/값 쌍입니다.

```
PresentationForm WC/PLSOutput
```

기본값은 PLSOutput입니다. 사용자가 WC를 지정하면 생성되는 중간 코드는 넓은 문자입니다. CTL 인쇄의 경우 이 기본값이 사용되어야 합니다.

로캘이 비CTL 로캘이고 키보드 값 PLSOutput을 가지면 해당 값은 무시되고 대신 mp(1)이 넓은 문자 코드를 생성합니다.

로캘이 CTL을 지원하는 경우 다음 표에 나열된 선택적 키워드/값 쌍을 사용할 수 있습니다. 이러한 변수는 테이블의 중간 열에 주어진 모든 가능한 값을 가정할 수 있습니다.

표 7-1 선택적 키워드/값 쌍

선택적 키워드	선택적 값	기본값
방향	ORIENTATION_LTR/	ORIENTATION_LTR
	ORIENTATION_RTL/	
	ORIENTATION_CONTEXTUAL	
숫자	NUMERALS_NOMINAL/	NUMERALS_NOMINAL
	NUMERALS_NATIONAL/	
	NUMERALS_CONTEXTUAL	
TextShaping	TEXT_SHAPED/	TEXT_SHAPED
	TEXT_NOMINAL/	
	TEXT_SHFORM1/	
	TEXT_SHFORM2/	
	TEXT_SHFORM3/ TEXT_SHFORM4	

## 프린터 상주 글꼴 추가하기

다음 예제는 구성 파일에 새 PCF, TrueType 또는 Type1 프린터 상주 글꼴을 추가할 때 따라야 하는 단계를 보여줍니다.

범위 0x00000021 - 0x0000007f에서 문자를 표시하기 위한 글꼴을 현재 구성된 PCF 글꼴 대신 트루타입 글꼴로 바꿉니다.

새 글꼴을 추가하기 전에 다음과 같이 현재 구성된 글꼴에 해당하는 구성 파일에서 다양한 구성 요소를 찾으십시오.

```

FontNameAlias iso88591R PCF /usr/openwin/lib/X11/fonts/75dpi/courR18PCF.Z
FontNameAlias iso88591B PCF /usr/openwin/lib/X11/fonts/75dpi/courB18PCF.Z
.
.
FontGroup iso88591 PCF iso88591R iso88591B
.
.
MapCode2Font 0x00000020 0x0000007f iso88591
.
.
CnvCode2Font iso88591R _xuiso88591 /usr/lib/lp/locale/$LANG/mp/xuiso88591.so
CnvCode2Font iso88591B _xuiso88591 /usr/lib/lp/locale/$LANG/mp/xuiso88591.so

```

en\_US.UTF-8 로캘에서 매핑을 수행하기 위한 후보로

/usr/openwin/lib/locale/ja/X11/fonts/TT/HG-MinchoL.ttf를 선택했다고 가정합니다. 이것은 .so 모듈 내의 매핑 함수에서 유니코드 문자 매핑된 트루타입 글꼴 파일이기 때문에 받는 UCS-2 코드 포인트를 직접 반환하는 함수만 가지면 됩니다.

```

unsigned short _ttfjis0201(unsigned short ucs2) {
    return(ucs2);
}

```

이것을 ttfjis0201.c 파일에 저장하십시오. 다음과 같이 공유 객체를 만드십시오.

```
cc -G -Kpic -o ttfjis0201.so ttfjis0201.c
```

/usr/openwin/lib/locale/ja/X11/fonts/75dpi/gotmrk20.pcf.Z 와 같은 PCF 파일을 매핑하고 있다면

/usr/openwin/lib/locale/ja/X11/fonts/75dpi/ 디렉토리에서 fonts.dir 파일을 찾으십시오. 다음과 같은 XLFD에 해당하는 인코딩에 익숙해 지십시오.

```
-sun-gothic-medium-r-normal--22-200-75-75-c-100-jisx0201.1976-0
```

jisx0201은 인코딩이기에 UCS-2에서 jisx0201로 매핑되는 공유 객체를 준비하십시오. .so 모듈을 만드는 매핑 테이블을 얻어야 합니다(이미 준비되지 않은 경우). 유니코드 로캘의 경우 ftp.unicode.org/pub/MAPPINGS/ 하에서 많은 문자 세트와 유니코드의 매핑을 찾으십시오. xu2jis0201.c 파일을 작성하기 위해 다음 매핑(1)(1)을 따르십시오.

```

unsigned short _xu2jis0201(unsigned short ucs2) {
    if(ucs2 >= 0x20 && ucs2 <= 0x7d )
        return (ucs2);
    if(ucs2==0x203e)
        return (0x7e);
    if(ucs2 >= 0xff61 && ucs2 <= 0xff9f)
        return (ucs2 - 0xff60 + 0xa0);
    return(0);
}

```

매핑 파일을 작성할 때 모든 UCS-2를 jisx0201 케이스에 포함시키십시오.

```
cc -G -o xu2jis0201.so xu2jis0201.c
```

## 공유 객체 파일 작성

이 예제에서 공유 객체 파일을 작성합니다.

다음 행을 `mp.conf`의 해당 부분에 추가하여 이 글꼴을 추가합니다. 다음 예제는 트루타입 글꼴을 추가하는 방법을 보여줍니다. PCF 글꼴은 트루타입 대신 PCF에 대한 키워드를 변경하는 것을 제외하고 동일한 패턴을 따릅니다.

```
FontNameAlias jis0201R TrueType /home/fn/HG-Minchol.ttf
FontGroup     jis0201 TrueType jis0201R
MapCode2Font 0x0020      0x007f jis0201
CnvCode2Font  jis0201R      _ttfjis0201 <.so path>
```

여기서 `.so` 경로는 `xu2jis0201.so` 파일을 가리킵니다.

변경된 `mp.conf` 파일로 `mp(1)`을 호출하면 범위 `0x0020-0x007f`가 새 글꼴로 인쇄됩니다. 다른 일본어 문자 범위도 동일하게 매핑합니다. `.so` 파일, 예를 들어, 범위 `0x0000FF61 0x0000FF9F`입니다.

역방향 호환성을 유지하려면

`/usr/openwin/lib/locale/$LANG/print/prolog.ps` 파일이 현재 로캘에서 출력을 만드는 데 사용됩니다. 현재 로캘에서 `$LANG`은 로캘 구성 요소 중 하나입니다. 그러한 상황에서 구성 파일 체계는 사용되지 않습니다.

예제 `mp.conf` 파일인 `/usr/lib/lp/locale/en_US.UTF-8/mp/mp.conf`를 참조하십시오.

---

## 기존 프롤로그 파일 사용자 정의 및 새 프롤로그 파일 추가하기

프롤로그 파일은 다음 두 가지 범주로 나눌 수 있습니다.

- 포스트스크립트 프롤로그 파일 (`.ps`)
- X 인쇄 서버 클라이언트 프롤로그 파일 (`.xpr`)

### 포스트스크립트 파일 사용자 정의

포스트스크립트 파일은 다음 범주 중 하나입니다.

- 공통 프롤로그 파일
- 인쇄 레이아웃 프롤로그 파일

## 로컬 종속 프롤로그 파일

prolog.ps 파일은 흔히 사용되지 않는 글꼴을 설정하기 위해 사용됩니다. 응용프로그램은 인쇄를 위해 미리 정의된 포스트스크립트 글꼴 이름을 사용합니다. 프롤로그 파일은 데스크 세트 캘린더 관리자 및 mp에 대하여 최소한 다음 글꼴 이름을 정의해야 합니다.

- LC\_Times-Roman
- LC\_Times-Bold
- LC\_Helvetica
- LC\_Helvetica-Bold
- LC\_Courier
- LC\_Helvetica-BoldOblique
- LC\_Times-Italic

다음 예제는 지정된 특정한 로컬 문자를 인쇄하기 위해 다음 글꼴을 사용합니다.

```
100 100 moveto
/LC_Times-Roman findfont 24 scale font setfont
(Any text string in your locale) show
```

Solaris 지역화 키트는 일본어 환경에 대해 예제 prolog.ps 파일을 제공합니다. 대신, 이 파일은 /usr/openwin/lib/locale/ja/print/ 디렉토리에 있습니다.

다음 예제는 기존 prolog.ps에서 복합 글꼴을 추가 또는 변경하는 방법을 보여줍니다.

```
%
(Foo-Fine) makecodeset12
(Base-Font) makeEUCfont
%
```

LC\_Base-Font라는 조합 글꼴을 정의하려 한다고 가정합니다.

LC\_Base-Font는 Foo-Fine의 조합 글꼴이며 Base-Font라는 기반 글꼴입니다. Foo-Fine은 로컬 문자 세트를 포함하는 글꼴입니다. 글꼴을 추가 또는 변경하기 위해서는 심층적인 포스트스크립트 지식이 필요 없습니다.

prolog.ps 파일을 작성하는 최선의 방법은 예제 버전을 연구하는 것입니다. prolog.ps 예제에서 두 개의 루틴 makecodeset12 및 makeEUCfont을 작성해야 합니다. makecodeset12는 로컬 글꼴 인코딩 정보를 설정합니다. 이 루틴은 로컬에 따라 다릅니다. makeEUCfont는 기반 글꼴과 로컬 글꼴을 조합하여 복합 글꼴을 형성합니다. 프롤로그 파일의 작성자는 makecodeset12 및 makeEUCfont를 작성하기 위해 포스트스크립트를 잘 알고 있어야 합니다.

prolog.ps 파일 지원은 역방향 호환성을 위해서만 유지됩니다. 로컬의 인쇄 필요를 위해 새 prolog.ps 파일을 작성하지 마십시오. 대신 mp.conf를 사용하십시오.

prolog.ps의 경로는 다음과 같습니다.

```
/usr/openwin/lib/locale/$LANG/print/prolog.ps
```





■ “Article from” - 입력이 새 그룹의 항목일 때

*Subject*                   우편 및 뉴스 헤더에서 취해진 주제 - s 옵션을 사용하여 일반 텍스트 파일은 물론 우편 및 뉴스 파일에 주제를 강제할 수 있습니다.

*Timenow*                   헤더와 푸터에 나타나는 인쇄 시간 이 정보는 localtime () 함수에서 취해집니다.

다음 함수는 프롤로그 파일의 인쇄 레이아웃에서 구현됩니다. 이러한 모든 함수는 부속 함수를 사용할 수 있습니다.

*endpage*                    사용 : page\_number endpage

인쇄된 페이지의 맨아래에 도달하면 호출됩니다. 이 함수는 페이지의 그래픽 컨텍스트를 복원하고 “showpage.”를 발행합니다. 일부 프롤로그 파일에서 헤더 및 푸터 정보는 컬럼별 모드보다 페이지별 모드에서만 표시됩니다. 이 함수를 구현하여 헤더 및 푸터 회색조 마름모꼴을 표시하는 부속 함수를 호출할 수 있습니다.

*newpage*                    사용 : page\_number newpage

새 페이지가 시작할 때 실행될 루틴 또는 명령 수직 인쇄 모드 설정, 인쇄 그래픽 컨텍스트 저장 및 페이지 좌표 변환은 루틴 기능의 일부입니다.

*endcol*                     사용 : page\_number col\_number endcol

헤더 및 푸터 표시에 사용됩니다. 새 인쇄 위치로의 이동 등입니다.

새 인쇄 레이아웃 프롤로그 파일 추가를 위해 인쇄 레이아웃 프롤로그 파일 내부에서 다음 변수를 정의해야 합니다.

*NumCols*                  인쇄 페이지의 열 수 기본값은 2입니다.

*PrintWidth*               인치 단위 인쇄 영역 너비 기본값은 6입니다.

*PrintHeight*              인치 단위 인쇄 영역 높이 기본값은 9입니다.

## .xpr File Customization

파일은 기본적으로 /usr/lib/lp/locale/C/mp/ 에 위치합니다. .xpr 파일은 mp.common.ps를 제외하고 각 포스트스크립트 프롤로그 레이아웃 파일에 해당합니다. MP\_PROLOGUE 환경 변수를 정의하여 대체 프롤로그 디렉토리를 정의할 수 있습니다.

파일은 키워드/값 쌍으로 작동합니다. #로 시작하는 행은 주석으로 간주됩니다. 공백은 명시적으로 진술된 경우를 제외하고 다른 토큰을 분리합니다. 각 .xpr 파일의 세 개의 주 부분은 다음 키워드 쌍으로 묶입니다.

- STARTCOMMON/ENDCOMMON
- STARTPAGE/ENDPAGE
- STARTCOLUMN/ENDCOLUMN
- STARTFORCEDPAGE/ENDFORCEDPAGE
- STARTFORCEDCOLUMN/ENDFORCEDCOLUMN

특정 키워드/값 쌍은 세 영역에서 사용됩니다. 각 영역은 다음에 설명됩니다.

## STARTCOMMON/ENDCOMMON 키워드

STARTCOMMON 키워드 뒤, ENDCOMMON 키워드 앞에 나타나는 모든 키워드/값 쌍은 인쇄 페이지의 일반 등록 정보를 정의합니다. 키워드의 다른 유효한 값은 "/"를 사용하여 구분됩니다.

ORIENTATION 0/1

"0"은 수직으로 인쇄가 이루어짐을 뜻하고 "1"은 수평으로 이루어짐을 뜻합니다.

PAGELength **부호없는 정수**

논리적 페이지당 행 수를 나타내는 값

LINELENGTH **부호없는 정수**

행당 단일 열 문자 수를 나타내는 값

NUMCOLS **부호없는 정수**

물리적 페이지당 논리적 페이지 수

HDNGFONTSIZE **부호없는 정수**

10포인트 단위의 머리글 글꼴 포인트 크기

BODYFONTSIZE **부호없는 정수**

10포인트 단위의 본문 글꼴 포인트

PROLOGDPI **부호없는 정수**

현재 .xpr 파일이 작성된 인치당 도트 눈금

YTEXTBOUNDARY **부호없는 정수**

y 좌표는 페이지 또는 논리적 페이지(열)의 텍스트 인쇄 경계를 설정합니다. 경계는 텍스트 인쇄가 예상 영역 내에 있는지 확인하는 추가 검사로 사용됩니다. 경계는 해당 글꼴에서 얻은 문자 높이 정보가 잘못되어있기 때문에 복잡한 텍스트 레이아웃 및 EUC 인쇄에 필요합니다.

STARTTEXT **부호없는 정수 부호없는 정수**

물리적 페이지의 첫번째 논리적 페이지에서 실제 텍스트 인쇄가 시작되는 10포인트 x/y 포인트

PAGESTRING 0/1

1은 "페이지" 문자열이 머리글의 페이지 수 전에 추가되어야 함을 나타냅니다.

0은 페이지 수만 표시됨을 나타냅니다.

#### EXTRAHDNGFONT 글꼴 문자열 1, 글꼴 문자열 2, ... 글꼴 문자열 n

'글꼴 문자열 n'에 대한 '글꼴 문자열 1'은 X 논리적 글꼴 설명입니다. 쉘표로 구분된 글꼴 목록의 EXTRAHDNGFONT는 공백이나 탭이 아닌 "입니다. 글꼴은 머리글이 인쇄될 때 내장 글꼴에 대한 환경 설정이 주어집니다. 일반적으로, EXTRABODYFONT는

```
/usr/openwin/server/etc/XpConfig/C/print/models/<model  
name>/fonts 디렉토리에 구성된 프린터 상주 글꼴을 할당하는데 사용됩니다.
```

fonts.dir 파일은 프린터 상주 글꼴의 XLFD를 포함합니다.

일반적으로 글꼴은 다음으로 지정됩니다.

```
"-monotype-Gill Sans-Regular-r-normal- *-%d-*-*p-0-iso8859-2"
```

.xpr 파일에서, "%d"가 있을 경우 mp(1)이 .xpr 파일의 현재 머리글 글꼴의 포인트 크기로 바뀝니다. x 해상도 및 y 해상도는 \*에 의해 지정되고 평균 너비 필드는 0으로 설정되어 조정 가능 글꼴의 선택을 나타냅니다. 더 많은 글꼴 이름을 부여할 수도 있습니다.

#### EXTRABODYFONT 글꼴 문자열 1, 글꼴 문자열 2, ... 글꼴 문자열 n

글꼴이 페이지 본문 인쇄에 사용된다는 것을 제외하고 EXTRAHDNGFONT와 동일합니다.

#### XDISPLACEMENT 부호있는/부호없는 정수

x 방향으로 페이지 내용을 이동하기 위해 페이지에 적용될 x 좌표 변위를 부여합니다. 이 변위는 +ve 또는 -ve 값일 수 있습니다.

#### YDISPLACEMENT 부호있는/부호없는 정수

이동이 y 방향으로 일어난다는 것을 제외하고 x 변위와 동일합니다.

두 키워드는 일부 프린터에 비표준 여백 너비를 찾고 페이지에서 인쇄된 내용을 이동해야 할 때 유용합니다.

## STARTPAGE/ENDPAGE

이 부분의 키워드 값 쌍은 STARTPAGE 및 ENDPAGE 키워드로 묶입니다. 이 부분은 물리적 페이지에 적용될 그리기 및 머리글 정보를 포함합니다. 물리적 페이지는 많은 논리적 페이지를 포함할 수 있지만 키워드 사이에 포함되는 모든 그리기 루틴은 물리적 페이지에 한 번만 적용됩니다.

유효한 그리기 엔티티는 LINE 및 ARC입니다. XDrawLine() 및 XDrawArc() 함수는 키워드 값에서 실행됩니다.

이 부분 내의 크기는 PROLOGDPI 단위로 매핑됩니다. 각도는 도 단위입니다.

LINE x1 y1 x2 y2      x/y 부호없는 좌표는 선 연결을 위한 포인트 쌍을 정의합니다.

ARC x y 너비 높이 각도1      x 및 y는 모두 원호 원점을 표현하는 부호없는 정수입니다. 너비 및 높이는 원호의 너비와 높이를 표현하는 부호없는 정수입니다.

USERSTRINGPOS x y	부호없는 좌표는 사용자 정보가 머리글에 인쇄되는 위치를 표시합니다.
TIMESTRINGPOS x y	부호없는 좌표는 인쇄 시간이 머리글에 인쇄되는 위치를 표시합니다.
PAGESTRINGPOS x y	부호없는 좌표는 각 인쇄된 페이지에 대한 페이지 문자열을 인쇄하는 위치를 표시합니다.
SUBJECTSTRINGPOS x y	부호없는 좌표는 페이지의 주제를 인쇄하는 위치를 표시합니다.

## STARTFORCEDPAGE/ENDFORCEDPAGE

-n 옵션이 mp에 주어지면 STARTPAGE/ENDPAGE 부분 내에서 모든 장식이 주어집니다. 그러나 -n 옵션이 주어져도 STARTFORCEDPAGE/ENDFORCEDPAGE 부분 내에 포함된 모든 것이 인쇄됩니다.

## STARTCOLUMN/ENDCOLUMN

모든 키워드는 이 부분의 항목이 물리적 페이지에 대한 NUMCOLS 횟수에 적용된다는 것을 제외하고 "STARTPAGE/ENDPAGE" 156 페이지의 설명과 동일합니다.

NUMCOLS이 3이면 물리적 페이지의 인쇄 가능 영역이 셋으로 분할되고 선, 원호 또는 머리글 장식이 페이지당 세 번 나타납니다.

## STARTFORCEDCOLUMN/ENDFORCEDCOLUMN

-n 옵션이 mp에 주어지면 STARTCOLUMN/ENDCOLUMN 부분 내에서 주어진 모든 장식이 인쇄되지 않습니다. 그러나 -n 옵션이 주어지지더라도 STARTFORCEDCOLUMN/ENDFORCEDCOLUMN 부분 내에 포함된 모든 것이 인쇄됩니다.

## 새 .xpr 파일 작성하기

다음 값은 이러한 값이 STARTCOMMON/ENDCOMMON 부분에 대한 .xpr 파일에 지정되지 않더라도 다른 키워드에 대한 mp(1) 프로그램 기본값입니다.

```
ORIENTATION 0
PAGELENGTH 60
LINELENGTH 80
YTEXTBOUNDARY 3005
NUMCOLS 01
```

```
HDNGFONTSIZE 120
PROLOGDPI 300
STARTTEXT 135 280
PAGESTRING 0
```

STARTPAGE/ENDPAGE 및 STARTCOLUMN/ENDCOLUMN에 묶이는 다른 두 부분을 위해 기본값이 필요하지 않습니다.

새 .xpr 프롤로그 파일을 작성할 때 기본값과 다른 값만 지정해야 합니다.

장식이 없는 페이지를 작성하려면 세로 형식으로 물리적 페이지당 네 개의 논리적 페이지를 사용합니다.

- STARTCOMMON
- NUMCOLS 04
- LINELENGTH 20
- ENDCOMMON

이 상황에서 다른 두 부분이 필요하지 않습니다.

- STARTPAGE/ENDPAGE
- STARTCOLUMN/ENDCOLUMN

매개 변수는 인쇄된 페이지에 장식을 두지 않을 때 필요합니다. 모든 좌표는 PROLOGDPI 키워드를 지정하지 않는 한 300dpi 기본값 범위입니다. 대상 프린터 해상도가 다르면 .xpr 파일은 프로그램에 의해 해당 해상도에 맞도록 조정됩니다.

.xpr 파일을 작성할 때 먼저 용지 크기를 알아야 합니다. 미국 용지의 경우 해상도 300dpi 프린터에 대하여 8.5x11인치, 2550X3300이 전체 크기입니다. 대부분의 프린터는 용지의 맨 위 왼쪽 모서리부터 인쇄되지 않습니다. 대신, 물리적 용지 주위에 여백을 둡니다. 이것은 0,0에서 인쇄를 하려해도 인쇄는 페이지의 맨 위 왼쪽 모서리에서 이루어지지 않음을 뜻합니다. 새 .xpr 파일을 작성할 때 이 제한을 고려해야 합니다.

## 부록 A

### iconv 코드 변환

다음 표는 Solaris 9 환경에서 사용 가능한 유니코드 관련 코드 변환 모듈을 나열합니다.

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈

원본 코드(기호)	대상 코드(기호)
646 (ISO 646)	UCS-2
646 (ISO 646)	USC-2BE
646 (ISO 646)	UCS-2LE
646 (ISO 646)	USC-4
646 (ISO 646)	USC-4BE
646 (ISO 646)	USC-4LE
646 (ISO 646)	UTF-8
646 (ISO 646)	UTF-16
646 (ISO 646)	UTF-16BE
646 (ISO 646)	UTF-16LE
646 (ISO 646)	UTF-32
646 (ISO 646)	UTF-32BE
646 (ISO 646)	UTF-32LE
ISO8859-11	UTF-8
8859-1 (ISO8859-1)	UCS-2
8859-1 (ISO8859-1)	UCS-2BE
8859-1 (ISO8859-1)	UCS-2LE

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
8859-1 (ISO8859-1)	UCS-4
8859-1 (ISO8859-1)	UCS-4BE
8859-1 (ISO8859-1)	UCS-4LE
8859-1 (ISO8859-1)	UTF-8
8859-1 (ISO8859-1)	UTF-16
8859-1 (ISO8859-1)	UTF-16BE
8859-1 (ISO8859-1)	UTF-16LE
8859-1 (ISO8859-1)	UTF-32
8859-1 (ISO8859-1)	UTF-32BE
8859-1 (ISO8859-1)	UTF-32LE
8859-2 (ISO8859-2)	UCS-2
8859-2 (ISO8859-2)	UCS-2BE
8859-2 (ISO8859-2)	UCS-2LE
8859-2 (ISO8859-2)	UCS-4
8859-2 (ISO8859-2)	UCS-4BE
8859-2 (ISO8859-2)	UCS-4LE
8859-2 (ISO8859-2)	UTF-8
8859-2 (ISO8859-2)	UTF-16
8859-2 (ISO8859-2)	UTF-16BE
8859-2 (ISO8859-2)	UTF-16LE
8859-2 (ISO8859-2)	UTF-32
8859-2 (ISO8859-2)	UTF-32BE
8859-2 (ISO8859-2)	UTF-32LE
8859-3 (ISO8859-3)	UCS-2
8859-3 (ISO8859-3)	UCS-2BE
8859-3 (ISO8859-3)	UCS-2LE
8859-3 (ISO8859-3)	UCS-4
8859-3 (ISO8859-3)	UCS-4BE



표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
8859-3 (ISO8859-3)	UCS-4LE
8859-3 (ISO8859-3)	UTF-8
8859-3 (ISO8859-3)	UTF-16
8859-3 (ISO8859-3)	UTF-16BE
8859-3 (ISO8859-3)	UTF-16LE
8859-3 (ISO8859-3)	UTF-32
8859-3 (ISO8859-3)	UTF-32BE
8859-3 (ISO8859-3)	UTF-32LE
8859-4 (ISO8859-4)	UCS-2
8859-4 (ISO8859-4)	UCS-2BE
8859-4 (ISO8859-4)	UCS-2LE
8859-4 (ISO8859-4)	UCS-4
8859-4 (ISO8859-4)	UCS-4BE
8859-4 (ISO8859-4)	UCS-4LE
8859-4 (ISO8859-4)	UTF-8
8859-4 (ISO8859-4)	UTF-16
8859-4 (ISO8859-4)	UTF-16BE
8859-4 (ISO8859-4)	UTF-16LE
8859-4 (ISO8859-4)	UTF-32
8859-4 (ISO8859-4)	UTF-32BE
8859-4 (ISO8859-4)	UTF-32LE
8859-5 (ISO8859-5)	UCS-2
8859-5 (ISO8859-5)	UCS-2BE
8859-5 (ISO8859-5)	UCS-2LE
8859-5 (ISO8859-5)	UCS-4
8859-5 (ISO8859-5)	UCS-4BE
8859-5 (ISO8859-5)	UCS-4LE
8859-5 (ISO8859-5)	UTF-8

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
8859-5 (ISO8859-5)	UTF-16
8859-5 (ISO8859-5)	UTF-16BE
8859-5 (ISO8859-5)	UTF-16LE
8859-5 (ISO8859-5)	UTF-32
8859-5 (ISO8859-5)	UTF-32BE
8859-5 (ISO8859-5)	UTF-32LE
8859-6 (ISO8859-6)	UCS-2
8859-6 (ISO8859-6)	UCS-2BE
8859-6 (ISO8859-6)	UCS-2LE
8859-6 (ISO8859-6)	UCS-4
8859-6 (ISO8859-6)	UCS-4BE
8859-6 (ISO8859-6)	UCS-4LE
8859-6 (ISO8859-6)	UTF-8
8859-6 (ISO8859-6)	UTF-16
8859-6 (ISO8859-6)	UTF-16BE
8859-6 (ISO8859-6)	UTF-16LE
8859-6 (ISO8859-6)	UTF-32
8859-6 (ISO8859-6)	UTF-32BE
8859-6 (ISO8859-6)	UTF-32LE
8859-7 (ISO8859-7)	UCS-2
8859-7 (ISO8859-7)	UCS-2BE
8859-7 (ISO8859-7)	UCS-2LE
8859-7 (ISO8859-7)	UCS-4
8859-7 (ISO8859-7)	UCS-4BE
8859-7 (ISO8859-7)	UCS-4LE
8859-7 (ISO8859-7)	UTF-8
8859-7 (ISO8859-7)	UTF-16
8859-7 (ISO8859-7)	UTF-16BE

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
8859-7 (ISO8859-7)	UTF-16LE
8859-7 (ISO8859-7)	UTF-32
8859-7 (ISO8859-7)	UTF-32BE
8859-7 (ISO8859-7)	UTF-32LE
8859-8 (ISO8859-8)	UCS-2
8859-8 (ISO8859-8)	UCS-2BE
8859-8 (ISO8859-8)	UCS-2LE
8859-8 (ISO8859-8)	UCS-4
8859-8 (ISO8859-8)	UCS-4BE
8859-8 (ISO8859-8)	UCS-4LE
8859-8 (ISO8859-8)	UTF-8
8859-8 (ISO8859-8)	UTF-16
8859-8 (ISO8859-8)	UTF-16BE
8859-8 (ISO8859-8)	UTF-16LE
8859-8 (ISO8859-8)	UTF-32
8859-8 (ISO8859-8)	UTF-32BE
8859-8 (ISO8859-8)	UTF-32LE
8859-9 (ISO8859-9)	UCS-2
8859-9 (ISO8859-9)	UCS-2BE
8859-9 (ISO8859-9)	UCS-2LE
8859-9 (ISO8859-9)	UCS-4
8859-9 (ISO8859-9)	UCS-4BE
8859-9 (ISO8859-9)	UCS-4LE
8859-9 (ISO8859-9)	UTF-8
8859-9 (ISO8859-9)	UTF-16
8859-9 (ISO8859-9)	UTF-16BE
8859-9 (ISO8859-9)	UTF-16LE
8859-9 (ISO8859-9)	UTF-32

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
8859-9 (ISO8859-9)	UTF-32BE
8859-9 (ISO8859-9)	UTF-32LE
8859-10 (ISO8859-10)	UCS-2
8859-10 (ISO8859-10)	UCS-2BE
8859-10 (ISO8859-10)	UCS-2LE
8859-10 (ISO8859-10)	UCS-4
8859-10 (ISO8859-10)	UCS-4BE
8859-10 (ISO8859-10)	UCS-4LE
8859-10 (ISO8859-10)	UTF-8
8859-10 (ISO8859-10)	UTF-16
8859-10 (ISO8859-10)	UTF-16BE
8859-10 (ISO8859-10)	UTF-16LE
8859-10 (ISO8859-10)	UTF-32
8859-10 (ISO8859-10)	UTF-32BE
8859-10 (ISO8859-10)	UTF-32LE
8859-13 (ISO8859-13)	UCS-2
8859-13 (ISO8859-13)	UCS-2BE
8859-13 (ISO8859-13)	UCS-2LE
8859-13 (ISO8859-13)	UCS-4
8859-13 (ISO8859-13)	UCS-4BE
8859-13 (ISO8859-13)	UCS-4LE
8859-13 (ISO8859-13)	UTF-8
8859-13 (ISO8859-13)	UTF-16
8859-13 (ISO8859-13)	UTF-16BE
8859-13 (ISO8859-13)	UTF-16LE
8859-13 (ISO8859-13)	UTF-32
8859-13 (ISO8859-13)	UTF-32BE
8859-13 (ISO8859-13)	UTF-32LE

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
8859-14 (ISO8859-14)	UCS-2
8859-14 (ISO8859-14)	UCS-2BE
8859-14 (ISO8859-14)	UCS-2LE
8859-14 (ISO8859-14)	UCS-4
8859-14 (ISO8859-14)	UCS-4BE
8859-14 (ISO8859-14)	UCS-4LE
8859-14 (ISO8859-14)	UTF-8
8859-14 (ISO8859-14)	UTF-16
8859-14 (ISO8859-14)	UTF-16BE
8859-14 (ISO8859-14)	UTF-16LE
8859-14 (ISO8859-14)	UTF-32
8859-14 (ISO8859-14)	UTF-32BE
8859-14 (ISO8859-14)	UTF-32LE
8859-15 (ISO8859-15)	UCS-2
8859-15 (ISO8859-15)	UCS-2BE
8859-15 (ISO8859-15)	UCS-2LE
8859-15 (ISO8859-15)	UCS-4
8859-15 (ISO8859-15)	UCS-4BE
8859-15 (ISO8859-15)	UCS-4LE
8859-15 (ISO8859-15)	UTF-8
8859-15 (ISO8859-15)	UTF-16
8859-15 (ISO8859-15)	UTF-16BE
8859-15 (ISO8859-15)	UTF-16LE
8859-15 (ISO8859-15)	UTF-32
8859-15 (ISO8859-15)	UTF-32BE
8859-15 (ISO8859-15)	UTF-32LE
8859-16 (ISO8859-16)	UCS-2
8859-16 (ISO8859-16)	UCS-2BE

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
8859-16 (ISO8859-16)	UCS-2LE
8859-16 (ISO8859-16)	UCS-4
8859-16 (ISO8859-16)	UCS-4BE
8859-16 (ISO8859-16)	UCS-4LE
8859-16 (ISO8859-16)	UTF-8
8859-16 (ISO8859-16)	UTF-16
8859-16 (ISO8859-16)	UTF-16BE
8859-16 (ISO8859-16)	UTF-16LE
8859-16 (ISO8859-16)	UTF-32
8859-16 (ISO8859-16)	UTF-32BE
8859-16 (ISO8859-16)	UTF-32LE
eucJP	UTF-8
gb2312	UTF-8
iso2022	UTF-8
ko_KR-cp933	UTF-8
ko_KR-euc	UTF-8
ko_KR-iso2022-7	UTF-8
ko_KR-johap	UTF-8
ko_KR-johap92	UTF-8
zh_TW-euc	UTF-8
zh_TW-cp937	UTF-8
zh_TW-iso2022-7	UTF-8
GBK	UTF-8
FujitsuJEF-ascii-code	UTF-8
FujitsuJEF-ascii-face	UTF-8
FujitsuJEF-kana-code	UTF-8
FujitsuJEF-kana-face	UTF-8
HitachiKEIS83	UTF-8

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
HitachiKEIS90	UTF-8
ISO-2022-JP	UTF-8
KOI8-R	UCS-2
KOI8-R	UCS-2BE
KOI8-R	UCS-2LE
KOI8-R	UCS-4
KOI8-R	UCS-4BE
KOI8-R	UCS-4LE
KOI8-R	UTF-8
KOI8-R	UTF-16
KOI8-R	UTF-16BE
KOI8-R	UTF-16LE
KOI8-R	UTF-32
KOI8-R	UTF-32BE
KOI8-R	UTF-32LE
KOI8-U	UCS-2
KOI8-U	UCS-2BE
KOI8-U	UCS-2LE
KOI8-U	UCS-4
KOI8-U	UCS-4BE
KOI8-U	UCS-4LE
KOI8-U	UTF-8
KOI8-U	UTF-16
KOI8-U	UTF-16BE
KOI8-U	UTF-16LE
KOI8-U	UTF-32
KOI8-U	UTF-32BE
KOI8-U	UTF-32LE

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
NECJIPS	UTF-8
PCK	UTF-8
UCS-2	646 (ISO 646)
UCS-2	8859-1 (ISO8859-1)
UCS-2	8859-2 (ISO8859-2)
UCS-2	8859-3 (ISO8859-3)
UCS-2	8859-4 (ISO8859-4)
UCS-2	8859-5 (ISO8859-5)
UCS-2	8859-6 (ISO8859-6)
UCS-2	8859-7 (ISO8859-7)
UCS-2	8859-8 (ISO8859-8)
UCS-2	8859-9 (ISO8859-9)
UCS-2	8859-10 (ISO8859-10)
UCS-2	8859-13 (ISO8859-13)
UCS-2	8859-14 (ISO8859-14)
UCS-2	8859-15 (ISO8859-15)
UCS-2	8859-16 (ISO8859-16)
UCS-2	KOI8-R
UCS-2	KOI8-U
UCS-2	UCS-4
UCS-2	UCS-4BE
UCS-2	UCS-4LE
UCS-2	UTF-7
UCS-2	UTF-8
UCS-2BE	646 (ISO 646)
UCS-2BE	8859-1 (ISO8859-1)
UCS-2BE	8859-2 (ISO8859-2)
UCS-2BE	8859-3 (ISO8859-3)



표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
UCS-2BE	8859-4 (ISO8859-4)
UCS-2BE	8859-5 (ISO8859-5)
UCS-2BE	8859-6 (ISO8859-6)
UCS-2BE	8859-7 (ISO8859-7)
UCS-2BE	8859-8 (ISO8859-8)
UCS-2BE	8859-9 (ISO8859-9)
UCS-2BE	8859-10 (ISO8859-10)
UCS-2BE	8859-13 (ISO8859-13)
UCS-2BE	8859-14 (ISO8859-14)
UCS-2BE	8859-15 (ISO8859-15)
UCS-2BE	8859-16 (ISO8859-16)
UCS-2BE	KOI8-R
UCS-2BE	KOI8-U
UCS-2BE	UCS-4
UCS-2BE	UCS-4BE
UCS-2BE	UCS-4LE
UCS-2BE	UTF-8
UCS-2LE	646 (ISO 646)
UCS-2LE	8859-1 (ISO8859-1)
UCS-2LE	8859-2 (ISO8859-2)
UCS-2LE	8859-3 (ISO8859-3)
UCS-2LE	8859-4 (ISO8859-4)
UCS-2LE	8859-5 (ISO8859-5)
UCS-2LE	8859-6 (ISO8859-6)
UCS-2LE	8859-7 (ISO8859-7)
UCS-2LE	8859-8 (ISO8859-8)
UCS-2LE	8859-9 (ISO8859-9)
UCS-2LE	8859-10 (ISO8859-10)

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
UCS-2LE	8859-13 (ISO8859-13)
UCS-2LE	8859-14 (ISO8859-14)
UCS-2LE	8859-15 (ISO8859-15)
UCS-2LE	8859-16 (ISO8859-16)
UCS-2LE	KOI8-R
UCS-2LE	KOI8-U
UCS-2LE	UCS-4
UCS-2LE	UCS-4BE
UCS-2LE	UCS-4LE
UCS-2LE	UTF-8
UCS-2LE	UTF-32
UCS-2LE	UTF-32BE
UCS-2LE	UTF-32LE
UCS-4	646
UCS-4	8859-1 (ISO8859-1)
UCS-4	8859-2 (ISO8859-2)
UCS-4	8859-3 (ISO8859-3)
UCS-4	8859-4 (ISO8859-4)
UCS-4	8859-5 (ISO8859-5)
UCS-4	8859-6 (ISO8859-6)
UCS-4	8859-7 (ISO8859-7)
UCS-4	8859-8 (SO 8859-8)
UCS-4	8859-9 (ISO8859-9)
UCS-4	8859-10 (ISO8859-10)
UCS-4	8859-13 (ISO8859-13)
UCS-4	8859-14 (ISO8859-14)
UCS-4	8859-15 (ISO8859-15)
UCS-4	8859-16 (ISO8859-16)

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
UCS-4	KOI8-R
UCS-4	KOI8-U
UCS-4	UCS-2
UCS-4	UCS-2BE
UCS-4	UCS-2LE
UCS-4	UTF-7
UCS-4	UTF-8
UCS-4	UCS-16
UCS-4	UCS-16BE
UCS-4	UCS-16LE
UCS-4	UTF-32
UCS-4	UCS-32BE
UCS-4	UCS-32LE
UCS-4BE	646
UCS-4BE	8859-1 (ISO8859-1)
UCS-4BE	8859-2 (ISO8859-2)
UCS-4BE	8859-3 (ISO8859-3)
UCS-4BE	8859-4 (ISO8859-4)
UCS-4BE	8859-5 (ISO8859-5)
UCS-4BE	8859-6 (ISO8859-6)
UCS-4BE	8859-7 (ISO8859-7)
UCS-4BE	8859-8 (SO 8859-8)
UCS-4BE	8859-9 (ISO8859-9)
UCS-4BE	8859-10 (ISO8859-10)
UCS-4BE	8859-13 (ISO8859-13)
UCS-4BE	8859-14 (ISO8859-14)
UCS-4BE	8859-15 (ISO8859-15)
UCS-4BE	8859-16 (ISO8859-16)

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
UCS-4BE	KOI8-R
UCS-4BE	KOI8-U
UCS-4BE	UCS-2
UCS-4BE	UCS-2BE
UCS-4BE	UCS-2LE
UCS-4BE	UCS-8
UCS-4BE	UCS-16
UCS-4BE	UCS-16BE
UCS-4BE	UCS-16LE
UCS-4BE	UCS-32
UCS-4BE	UCS-32BE
UCS-4BE	UCS-32LE
UCS-4LE	646 (ISO 646)
UCS-4LE	8859-1 (ISO8859-1)
UCS-4LE	8859-2 (ISO8859-2)
UCS-4LE	8859-3 (ISO8859-3)
UCS-4LE	8859-4 (ISO8859-4)
UCS-4LE	8859-5 (ISO8859-5)
UCS-4LE	8859-6 (ISO8859-6)
UCS-4LE	8859-7 (ISO8859-7)
UCS-4LE	8859-8 (SO 8859-8)
UCS-4LE	8859-9 (ISO8859-9)
UCS-4LE	8859-10 (ISO8859-10)
UCS-4LE	8859-13 (ISO8859-13)
UCS-4LE	8859-14 (ISO8859-14)
UCS-4LE	8859-15 (ISO8859-15)
UCS-4LE	8859-16 (ISO8859-15)
UCS-4LE	KOI8-R

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
UCS-4LE	KOI8-U
UCS-4LE	UCS-2
UCS-4LE	UCS-2BE
UCS-4LE	UCS-2LE
UCS-4LE	UTF-16
UCS-4LE	UTF-16BE
UCS-4LE	UTF-16LE
UCS-4LE	UTF-8
UTF-7	UCS-2
UTF-7	UCS-4
UTF-7	UCS-8
UTF-8	646 (ISO 646)
UTF-8	8859-1 (ISO8859-1)
UTF-8	8859-2 (ISO8859-2)
UTF-8	8859-3 (ISO8859-3)
UTF-8	8859-4 (ISO8859-4)
UTF-8	8859-5 (ISO8859-5)
UTF-8	8859-6 (ISO8859-6)
UTF-8	8859-7 (ISO8859-7)
UTF-8	8859-8 (ISO8859-8)
UTF-8	8859-9 (ISO8859-9)
UTF-8	8859-10 (ISO8859-10)
UTF-8	8859-11 (ISO8859-11)
UTF-8	8859-13 (ISO8859-13)
UTF-8	8859-14 (ISO8859-14)
UTF-8	8859-15 (ISO8859-15)
UTF-8	8859-16 (ISO8859-16)
UTF-8	eucJP

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
UTF-8	gb2312
UTF-8	iso2022
UTF-8	ko_KR-euc
UTF-8	ko_KR-johap
UTF-8	ko_KR-johap92
UTF-8	ko_KR-iso2022-7
UTF-8	zh_TW-euc
UTF-8	zh_TW-iso2022-7
UTF-8	zh_TW-cp937
UTF-8	FujitsuJEF-ascii-code
UTF-8	FujitsuJEF-ascii-face
UTF-8	FujitsuJEF-kana-code
UTF-8	FujitsuJEF-kana-face
UTF-8	GBK
UTF-8	HitachiKEIS83
UTF-8	HitachiKEIS90
UTF-8	ISO-2022-JP
UTF-8	KOI8-R
UTF-8	KOI8-U
UTF-8	UTF-7
UTF-8	NECJIPS
UTF-8	PCK
UTF-8	UCS-2
UTF-8	UCS-2BE
UTF-8	UCS-2LE
UTF-8	UCS-4
UTF-8	UCS-4BE
UTF-8	UCS-4LE

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
UTF-8	UTF-7
UTF-8	UTF-16
UTF-8	UTF-16BE
UTF-8	UCS-16LE
UTF-16	646 (ISO 646)
UTF-16	8859-1 (ISO8859-1)
UTF-16	8859-2 (ISO8859-2)
UTF-16	8859-3 (ISO8859-3)
UTF-16	8859-4 (ISO8859-4)
UTF-16	8859-5 (ISO8859-5)
UTF-16	8859-6 (ISO8859-6)
UTF-16	8859-7 (ISO8859-7)
UTF-16	8859-8 (ISO8859-8)
UTF-16	8859-9 (ISO8859-9)
UTF-16	8859-10 (ISO8859-10)
UTF-16	8859-13 (ISO8859-13)
UTF-16	8859-14 (ISO8859-14)
UTF-16	8859-15 (ISO8859-15)
UTF-16	8859-15 (ISO8859-15)
UTF-16	8859-16 (ISO8859-16)
UTF-16	KOI8-R
UTF-16	KOI8-U
UTF-16	UCS-4
UTF-16	UCS-4BE
UTF-16	UCS-4LE
UTF-16	UTF-8
UTF-16BE	646 (ISO 646)
UTF-16BE	8859-1 (ISO8859-1)

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
UTF-16BE	8859-2 (ISO8859-2)
UTF-16BE	8859-3 (ISO8859-3)
UTF-16BE	8859-4 (ISO8859-4)
UTF-16BE	8859-5 (ISO8859-5)
UTF-16BE	8859-6 (ISO8859-6)
UTF-16BE	8859-7 (ISO8859-7)
UTF-16BE	8859-8 (ISO8859-8)
UTF-16BE	8859-9 (ISO8859-9)
UTF-16BE	8859-10(ISO8859-10)
UTF-16BE	8859-13 (ISO8859-13)
UTF-16BE	8859-14 (ISO8859-14)
UTF-16BE	8859-15 (ISO8859-15)
UTF-16BE	8859-16 (ISO8859-16)
UTF-16BE	KOI8-R
UTF-16BE	KOI8-U
UTF-16BE	UCS-4
UTF-16BE	UCS-4BE
UTF-16BE	UCS-4LE
UTF-16BE	UTF-8
UTF-16LE	646 (ISO 646)
UTF-16LE	8859-1 (ISO8859-1)
UTF-16LE	8859-2 (ISO8859-2)
UTF-16LE	8859-3 (ISO8859-3)
UTF-16LE	8859-4 (ISO8859-4)
UTF-16LE	8859-5 (ISO8859-5)
UTF-16LE	8859-6 (ISO8859-6)
UTF-16LE	8859-7 (ISO8859-7)
UTF-16LE	8859 -8 (ISO8859-8)



표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
UTF-16LE	8859-9 (ISO8859-9)
UTF-16LE	8859-10 (ISO8859-10)
UTF-16LE	8859-13 (ISO8859-13)
UTF-16LE	8859-14 (ISO8859-14)
UTF-16LE	8859-15 (ISO8859-15)
UTF-16LE	8859-16 (ISO8859-16)
UTF-16LE	KOI8-R
UTF-16LE	KOI8-U
UTF-16LE	UCS-4
UTF-16LE	UCS-4BE
UTF-16LE	UCS-4LE
UTF-16LE	UTF-8
UTF-32	UTF-8
UTF-32	UCS-2
UTF-32	UCS-2BE
UTF-32	UCS-2LE
UTF-32	UCS-4
UTF-32	UCS-4BE
UTF-32	UCS-4LE
UTF-32	UTF-16
UTF-32	UTF-16LE
UTF-32	UTF-32BE
UTF-32	646 (ISO 646)
UTF-32	ISO8859-1
UTF-32	ISO8859-2
UTF-32	ISO8859-3
UTF-32	ISO8859-4
UTF-32	ISO8859-5

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
UTF-32	ISO8859-6
UTF-32	ISO8859-7
UTF-32	ISO8859-8
UTF-32	ISO8859-9
UTF-32	ISO8859-10
UTF-32	ISO8859-13
UTF-32	ISO8859-14
UTF-32	ISO8859-15
UTF-32	ISO8859-16
UTF-32	KOI8-R
UTF-32	KOI8-U
UTF-32BE	UTF-8
UTF-32BE	UCS-2
UTF-32BE	UCS-2BE
UTF-32BE	UCS-2LE
UTF-32BE	UCS-4
UTF-32BE	UCS-4BE
UTF-32BE	UCS-4LE
UTF-32BE	UTF-16
UTF-32BE	UTF-16BE
UTF-32 BE	UTF-16LE
UTF-32BE	646 (ISO 646)
UTF-32BE	ISO8859-1
UTF-32BE	ISO8859-2
UTF-32BE	ISO8859-3
UTF-32BE	ISO8859-4
UTF-32BE	ISO8859-5
UTF-32BE	ISO8859-6

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
UTF-32BE	ISO8859-7
UTF-32BE	ISO8859-8
UTF-32BE	ISO8859-9
UTF-32BE	ISO8859-10
UTF-32BE	ISO8859-13
UTF-32BE	ISO8859-14
UTF-32BE	ISO8859-15
UTF-32BE	ISO8859-16
UTF-32BE	KOI8-R
UTF-32BE	KOI8-U
UTF-32LE	UTF-8
UTF-32LE	UCS-2
UTF-32LE	UCS-2BE
UTF-32LE	UCS-2LE
UTF-32LE	UCS-4
UTF-32LE	UCS-4BE
UTF-32LE	UCS-4LE
UTF32-LE	UTF-16
UTF32-LE	UTF-16BE
UTF-32LE	UTF-16LE
UTF-32LE	646 (ISO 646)
UTF-32LE	ISO8859-1
UTF-32LE	ISO8859-2
UTF-32LE	ISO8859-3
UTF-32LE	ISO8859-4
UTF-32LE	ISO8859-5
UTF-32LE	ISO8859-6
UTF-32LE	ISO8859-7

표 A-1 사용 가능한 유니코드 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
UTF-32LE	ISO8859-8
UTF-32LE	ISO8859-9
UTF-32LE	ISO8859-10
UTF-32LE	ISO8859-13
UTF-32LE	ISO8859-14
UTF-32LE	ISO8859-15
UTF-32LE	ISO8859-16
UTF-32LE	KOI8-R
UTF-32LE	KOI8-U

주 - UTF-EBCDIC은 새로운 IBM 코드 페이지 이름입니다. Solaris 9 환경은 양방향 UTF-8 <—> UTF-EBCDIC 변환도 지원합니다.

다음 표는 Solaris 9 환경에서 사용 가능한 유니코드 및 IBM/Microsoft EBCDIC 및 PC iconv 코드 변환 모듈을 나열합니다.

표 A-2 사용 가능한 유니코드 및 IBM/Microsoft EBCDIC 및 PC 코드 페이지 관련 iconv 코드 변환 모듈

원본 코드(기호)	대상 코드(기호)
UTF-8	IBM-0371
UTF-8	IBM-273
UTF-8	IBM-277
UTF-8	IBM-278
UTF-8	IBM-280
UTF-8	IBM-284
UTF-8	IBM-285
UTF-8	IBM-297
UTF-8	IBM-420
UTF-8	IBM-424

표 A-2 사용 가능한 유니코드 및 IBM/Microsoft EBCDIC 및 PC 코드 페이지 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
UTF-8	IBM-500
UTF-8	IBM-850
UTF-8	IBM-852
UTF-8	IBM-855
UTF-8	IBM-856
UTF-8	IBM-857
UTF-8	IBM-862
UTF-8	IBM-864
UTF-8	IBM-866
UTF-8	IBM-869
UTF-8	IBM-870
UTF-8	IBM-871
UTF-8	IBM-875
UTF-8	IBM-880
UTF-8	IBM-1025
UTF-8	IBM-1026
UTF-8	IBM-1112
UTF-8	IBM-1122
UTF-8	IBM-921
UTF-8	IBM-922
UTF-8	IBM-1046
UTF-8	IBM-1140
UTF-8	IBM-1141
UTF-8	IBM-1142
UTF-8	IBM-1143
UTF-8	IBM-1144
UTF-8	IBM-1145

표 A-2 사용 가능한 유니코드 및 IBM/Microsoft EBCDIC 및 PC 코드 페이지 관련 iconv 코드 변환 모듈 (계속)

원본 코드(기호)	대상 코드(기호)
UTF-8	IBM-1146
UTF-8	IBM-1147
UTF-8	IBM-1148
UTF-8	IBM-1149
UTF-8	CP850
UTF-8	CP852
UTF-8	CP855
UTF-8	CP857
UTF-8	CP862
UTF-8	CP864
UTF-8	CP866
UTF-8	CP869
UTF-8	CP874
UTF-8	CP1250
UTF-8	CP1251
UTF-8	CP1252
UTF-8	CP1253
UTF-8	CP1254
UTF-8	CP1255
UTF-8	CP1256
UTF-8	CP1257
UTF-8	CP1258

다음 표는 UTF-8에 대한 사용 가능한 iconv 코드 변환 IBM 및 Microsoft EBCDIC/PC 코드 페이지를 나열합니다.

표 A-3 UTF-8에 대한 사용 가능한 iconv 코드 변환 - IBM 및 Microsoft EBCDIC/PC 코드 페이지

UTF-EBCDIC	UTF-8
IBM-037	UTF-8

표 A-3 UTF-8에 대한 사용 가능한 iconv 코드 변환 - IBM 및 Microsoft EBCDIC/PC 코드 페이지 (계속)

UTF-EBCDIC	UTF-8
IBM-273	UTF-8
IBM-277	UTF-8
IBM-278	UTF-8
IBM-280	UTF-8
IBM-284	UTF-8
IBM-285	UTF-8
IBM-297	UTF-8
IBM-420	UTF-8
IBM-424	UTF-8
IBM-500	UTF-8
IBM-850	UTF-8
IBM-852	UTF-8
IBM-855	UTF-8
IBM-856	UTF-8
IBM-857	UTF-8
IBM-862	UTF-8
IBM-864	UTF-8
IBM-866	UTF-8
IBM-869	UTF-8
IBM-870	UTF-8
IBM-871	UTF-8
IBM-875	UTF-8
IBM-880	UTF-8
IBM-921	UTF-8
IBM-922	UTF-8
IBM-1025	UTF-8
IBM-1026	UTF-8
IBM-1046	UTF-8

표 A-3 UTF-8에 대한 사용 가능한 iconv 코드 변환 - IBM 및 Microsoft EBCDIC/PC 코드 페이지 (계속)

UTF-EBCDIC	UTF-8
IBM-1112	UTF-8
IBM-1122	UTF-8
IBM-1140	UTF-8
IBM-1141	UTF-8
IBM-1142	UTF-8
IBM-1143	UTF-8
IBM-1144	UTF-8
IBM-1145	UTF-8
IBM-1146	UTF-8
IBM-1147	UTF-8
IBM-1148	UTF-8
IBM-1149	UTF-8
CP850	UTF-8
CP852	UTF-8
CP855	UTF-8
CP857	UTF-8
CP862	UTF-8
CP864	UTF-8
CP866	UTF-8
CP869	UTF-8
CP874	UTF-8
CP1250	UTF-8
CP1251	UTF-8
CP1252	UTF-8
CP1253	UTF-8
CP1254	UTF-8
CP1255	UTF-8
CP1256	UTF-8



표 A-3 UTF-8에 대한 사용 가능한 iconv 코드 변환 - IBM 및 Microsoft EBCDIC/PC 코드 페이지 (계속)

UTF-EBCDIC	UTF-8
CP1257	UTF-8
CP1258	UTF-8



## 부록 B

# Software CD의 부분 로캘 패키지 목록

다음 표는 Software CD의 패키지 및 내용을 나열합니다.

표 B-1 부분 로캘의 목록

패키지 이름	설명
JSatsvr	일본어 Solaris용 일본어 입력 시스템 ATOK12 루트 파일
JSatsvu	일본어 Solaris용 일본어 입력 시스템 ATOK12 usr 파일
JSatsvw	일본어 Solaris용 일본어 입력 시스템 ATOK12 X11 지원 파일
NSCPcpcom	국제 보안 규정을 지원하는 Netscape Communicator 4.7의 간체 한자 부분 버전
NSCPhpcom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 정체 한자 부분 버전
NSCPjacom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 일본어 버전(공통)
NSCPkpcom	국제 보안 규정을 지원하는 Netscape Communicator 4.78 한국어 부분 버전
SUNW5ttf	정체 한자 트루 타입 글꼴 패키지
SUNWale	중국어, 일본어 및 한국어 로캘이 공유하는 공통 파일. 아시아권 언어 환경 실행을 위한 필수 패키지
SUNWaled	중국어, 일본어 및 한국어 로캘이 공유하는 설명서 페이지
SUNWalex	중국어, 일본어 및 한국어 로캘이 공유하는 공통 파일. 아시아권 언어 환경(64비트) 실행을 위한 필수 패키지
SUNWauadt	오스트랄라시아 CDE 지원
SUNWauaos	오스트랄라시아 OS 지원
SUNWauaow	오스트랄라시아 개방형 Windows 지원

표 B-1 부분 로케일의 목록 (계속)

패키지 이름	설명
SUNWauaox	오스트랄라시아 16비트 개방형 Windows 지원
SUNWcamdt	중미 CDE 지원
SUNWcamos	중미 OS 지원
SUNWcamow	중미 개방형 Windows 지원
SUNWcamox	중미 64비트 OS 지원
SUNWcdt	CDE 데스크탑 로그인 환경 간체 한자 버전
SUNWceudt	중부 유럽 CDE 지원
SUNWceuos	중부 유럽 OS 지원
SUNWceuow	중부 유럽 개방형 Windows 지원
SUNWceuox	중부 유럽 64비트 OS 지원
SUNWcleu	간체 한자(EUC) 언어 환경 특정 파일간체 한자(EUC) 언어 환경 실행을 위한 필수 패키지
SUNWcleux	간체 한자 언어 환경 특정 파일간체 한자(64비트) 언어 환경 실행을 위한 필수 패키지
SUNWcttf	간체 한자(EUC) 트루 타입 글꼴
SUNWcufnt	간체 한자(UTF-8) X 윈도우 플랫폼 필수 글꼴
SUNWcxplt	간체 한자 X 윈도우 플랫폼 소프트웨어 패키지
SUNWdelu	
SUNWdespl	맞춤법 검사 엔진 - 독일어 사전
SUNWeeudt	동유럽 CDE 지원
SUNWeeuos	동유럽 OS 지원
SUNWeeuow	동유럽 개방형 Windows 지원
SUNWeeuox	동유럽 64비트 OS 지원
SUNWeslu	라이브 업그레이드 L10N 스페인어 메시지 파일
SUNWesspl	맞춤법 검사 엔진 - 스페인어 사전
SUNWeu8df	미국 영어/UTF-8 코어 데스크탑 파일
SUNWeu8os	OS 환경 사용자 파일용 미국 영어/UTF-8 L10N
SUNWeu8ox	OS 환경 사용자 파일(64비트)용 미국 영어/UTF-8 L10N
SUNWeudba	CDE Base용 미국 영어/UTF-8 L10N

표 B-1 부분 로케일의 목록 (계속)

패키지 이름	설명
SUNWeudbd	CDE Dtbuilder용 미국 영어/UTF-8 L10N
SUNWeudda	CDE 데스크탑 응용프로그램용 미국 영어/UTF-8 L10N
SUNWeudhr	CDE 도움말 런타임용 미국 영어/UTF-8 L10N
SUNWeudhs	CDE 도움말 런타임용 미국 영어/UTF-8 L10N
SUNWeudis	CDE 아이콘용 미국 영어/UTF-8 L10N
SUNWeudiv	데스크탑 이미지 도구용 미국 영어/UTF-8 L10N
SUNWeudlg	CDE 데스크탑 로그인 환경용 미국 영어/UTF-8 L10N
SUNWeudmg	데스크탑 창 관리자용 미국 영어/UTF-8 L10N
SUNWeuez	Desktop Power Pack Applications용 미국 영어/UTF-8 L10N
SUNWeuhed	OS 환경 사용자 파일용 미국 영어/UTF-8 L10N
SUNWeuluf	환경 사용자 파일용 미국 영어/UTF-8 L10N
SUNWeulux	환경 사용자 파일(64비트)용 미국 영어/UTF-8 L10N
SUNWeusru	Solaris 사용자 등록용 미국 영어/UTF-8 L10N
SUNWfrlu	라이브 업그레이드 L10N 프랑스어 메시지 파일
SUNWfrspl	맞춤법 검사 엔진 - 프랑스어 사전
SUNWgttf	간체 한자(GBK) 트루 타입 글꼴
SUNWgxfnt	간체 한자(GBK) X 윈도우 플랫폼 필수 글꼴
SUNWgxplx	간체 한자(GBK) X 윈도우 시스템 플랫폼 소프트웨어 패키지(64비트)
SUNWhdt	CDE 데스크탑 로그인 환경용 정체 한자 지역화
SUNWhkdt	CDE 데스크탑 로그인 환경 정체 한자(홍콩) 버전
SUNWhkfnt	정체 한자 BIG5 (홍콩) X 윈도우 플랫폼 필수 패키지
SUNWhkleu	정체 한자(홍콩) 언어 환경 특정 파일
SUNWhklex	정체 한자(홍콩) 언어 환경 사용자 파일(64비트)
SUNWhkplt	정체 한자(홍콩) X 윈도우 플랫폼 소프트웨어 패키지
SUNWhkplx	정체 한자(홍콩) X 윈도우 플랫폼 소프트웨어 패키지(64비트)
SUNWhkttf	정체 한자 홍콩 추가 문자 세트 트루 타입 글꼴 패키지.
SUNWhleu	정체 한자 언어 환경 특정 파일정체 한자 언어 환경 실행을 위한 필수 패키지

표 B-1 부분 로캘의 목록 (계속)

패키지 이름	설명
SUNWhleux	정체 한자(EUC) 언어 환경 특정 파일 정체 한자 언어 환경(64비트) 실행을 위한 필수 패키지
SUNWhttf	정체 한자 트루 타입 글꼴 패키지
SUNWhufnt	간체 한자(UTF-8) X 윈도우 플랫폼 필수 글꼴
SUNWhxfnt	정체 한자 X 윈도우 시스템 플랫폼 필수 글꼴 패키지
SUNWhxplt	정체 한자 X 윈도우 시스템 플랫폼 소프트웨어 패키지
SUNWi1cs	X11 ISO8859-1 코드 세트 지원
SUNWi2cs	X11 ISO8859-2 코드 세트 지원
SUNWi2of	ISO-8859-2 문자 세트(선택 글꼴)용 X11 글꼴
SUNWi5cs	X11 ISO8859-5 코드 세트 지원
SUNWi7cs	X11 ISO8859-7 코드 세트 지원
SUNWi9cs	X11 ISO8859-9 코드 세트 지원
SUNWi13cs	X11 ISO8859-13 코드 세트 지원
SUNWi15cs	X11 ISO8859-15 코드 세트 지원
SUNWiiimr	인터넷/인트라넷 입력 방식 프레임워크 (루트)
SUNWiiimu	인터넷/인트라넷 입력 방식 프레임워크 (Usr)
SUNWindt	CDE 데스크탑 로그인 환경 인도어 지역화
SUNWinfnt	인도어 (UTF-8) X 윈도우 플랫폼 필수 글꼴
SUNWinleu	이 패키지는 인도어 환경 특정 파일을 포함합니다.인도어 언어 환경 실행을 위한 필수 패키지
SUNWinlex	이 패키지는 인도어 환경 특정 파일을 포함합니다.인도어 언어 환경 실행을 위한 필수 패키지(64비트).
SUNWinplt	인도어 X 윈도우 시스템 플랫폼 소프트웨어
SUNWinttf	인도어 트루타입 글꼴
SUNWitlu	라이브 업그레이드 지역화 이탈리아어 메시지 파일
SUNWitspl	맞춤법 검사 엔진 - 이탈리아어 사전
SUNWj3jmp	J2SDK RELEASE 설명서 페이지 일본어 버전
SUNWjedt	CDE 데스크탑 로그인 환경 일본어(EUC) 버전
SUNWjexp	X 윈도우 시스템 플랫폼 소프트웨어(64비트) 일본어(EUC) 버전

표 B-1 부분 로케일의 목록 (계속)

패키지 이름	설명
SUNWjfprr	일본어 기능 패키지(JFP)용 스트림 모듈. JFP 환경 지원을 위한 필수 패키지
SUNWjfpur	usr용 일본어 기능 패키지(JFP) 특정 파일 JFP 환경 지원을 위한 필수 패키지
SUNWjfpux	usr용 일본어 기능 패키지(JFP) 지향 64비트 파일. JFP 환경 실행을 위한 필수 패키지
SUNWjman	SUNWjfprr과 SUNWjfpur용 영어 설명서 페이지를 보기 위한 일본어 기능 패키지 설명서 페이지
SUNWjulcf	일본어 UTF-8 로케일 환경 공동 파일
SUNWjxplt	X 윈도우 시스템 플랫폼 소프트웨어용 일본어 버전
SUNWkdt	CDE 데스크탑 로그인 환경 한국어 지역화
SUNWkleu	한국어 언어 환경 특정 파일. 한국어 언어 환경 실행을 위한 필수 패키지
SUNWkleux	한국어(EUC) 언어 환경 특정 파일 한국어 언어 환경(64비트) 실행을 위한 필수 패키지
SUNWkttf	한국어 트루 타입 글꼴
SUNWkxfnt	한국어 X 윈도우 플랫폼 필수 글꼴
SUNWkxplt	한국어 X 윈도우 플랫폼 소프트웨어 패키지
SUNWmeadt	중동 CDE 지원
SUNWmeaos	중동 OS 지원
SUNWmeaow	중동 OW 지원
SUNWmeaox	중동 64비트 OS 지원
SUNWnafos	북아프리카 OS 지원
SUNWnafow	북아프리카 OW 지원
SUNWnafox	북아프리카 64비트 OS 지원
SUNWneudt	북유럽 CDE 지원
SUNWneuos	북유럽 OS 지원
SUNWneuow	북유럽 OW 지원
SUNWneuox	북유럽 64비트 OS 지원
SUNWplow	부분 로케일용 OpenWindows 활성화
SUNWplow1	추가 부분 로케일용 OpenWindows 활성화

표 B-1 부분 로케일의 목록 (계속)

패키지 이름	설명
SUNWsamdt	남미 CDE 지원
SUNWsamos	남미 OS 지원
SUNWsamow	남미 OW 지원
SUNWsamox	남미 64비트 OS 지원
SUNWseudt	남부 유럽 CDE 지원
SUNWseuos	남부 유럽 OS 지원
SUNWseuow	남부 유럽 개방형 Windows 지원
SUNWseuox	남부 유럽 64비트 OS 지원
SUNWsvs	스웨덴어 설치 소프트웨어 지역화
SUNWsvlu	라이브 업그레이드 지역화 스웨덴어 메시지 파일
SUNWsvspl	맞춤법 검사 엔진 - 스웨덴어 사전
SUNWtdbas	CDE 데스크탑 로그인 환경 타이어 지역화
SUNWtddst	CDE 데스크탑 응용프로그램용 타이어 버전
SUNWtddte	CDE 데스크탑 로그인 환경 타이어 버전
SUNWtdft	CDE 글꼴용 타이어 버전
SUNWtdwm	CDE 데스크탑 창 관리자용 타이어 버전
SUNWtleu	타이어 언어 환경 특정 파일 타이어 언어 환경 실행을 위한 필수 패키지
SUNWtleux	타이어 언어 환경 특정 파일. 타이어 언어 환경(64비트) 실행을 위한 필수 패키지
SUNWtxplt	타이어 X 윈도우 플랫폼 소프트웨어 패키지
SUNWweudt	서유럽 CDE 지원
SUNWweuos	서유럽 OS 지원
SUNWweuow	서유럽 개방형 Windows 지원
SUNWweuox	서유럽 64비트 OS 지원



## 부록 C

# Language CD의 전체 로케일 패키지 목록

다음 표는 간체 한자 패키지와 내용을 나열합니다.

표 C-1 간체 한자

패키지 이름	설명
IPLTcadcon	관리 서버 콘솔 클라이언트 패키지 간체 한자 버전
IPLTcadman	관리 서버 설명서 간체 한자 버전
IPLTcadmin	관리 서버 간체 한자 버전
IPLTccons	콘솔 클라이언트 기반 패키지 간체 한자 버전
IPLTcdscon	디렉토리 서버 콘솔 간체 한자 버전
IPLTcdsman	디렉토리 서버 설명서 간체 한자 버전
IPLTcdsu	디렉토리 서버 간체 한자 버전
NSCPccom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 간체 한자 버전
NSCPcucom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 zh.UTF-8 버전
NSCPgcom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 zh.UTF-8 버전
SUNWcacx	간체 한자 AccessX 클라이언트 프로그램
SUNWcadis	Admintool과 GUI 설치용 간체 한자(EUC) 버전
SUNWcadma	시스템 관리 작업 수행을 위해 사용되는 소프트웨어용 간체 한자(EUC) 버전. Admintool은 이 패키지와 간체 한자(EUC) 버전용 SUNWhadis 패키지를 필요로 합니다.
SUNWcbcp	간체 한자(EUC) 언어 환경 이진 호환 파일
SUNWcdab	CDE Desktop Application Builder 간체 한자(EUC) 버전
SUNWcdbas	CDE Base 기능 간체 한자(EUC) 버전

표 C-1 간체 한자 (계속)

패키지 이름	설명
SUNWcdcl	Solaris 디스크 없는 클라이언트 관리 응용프로그램 간체 한자 버전.
SUNWcddst	CDE 데스크탑 응용프로그램 간체 한자(EUC) 버전
SUNWcddte	CDE 데스크탑 로그인 환경 간체 한자(EUC) 버전
SUNWcdez	Desktop Power Pack Applications 간체 한자(EUC) 버전
SUNWcdf	CDE 글꼴 간체 한자(EUC) 버전
SUNWcdhcm	DHCP Manager 간체 한자 버전
SUNWcdhe	CDE 도움말 런타임 환경 간체 한자(EUC) 버전
SUNWcdhev	간체 한자(EUC) CDE 도움말 볼륨
SUNWcdhez	간체 한자(EUC)(공통) Desktop Power Pack 도움말 볼륨
SUNWcdicn	CDE 아이콘 간체 한자(EUC) 버전
SUNWcdim	CDE 이미지 도구 간체 한자(EUC) 버전
SUNWcdwm	CDE 데스크탑 창 관리자 간체 한자(EUC) 버전
SUNWcepwm	Power Management OW 유틸리티 간체 한자(EUC) 버전
SUNWcfdl	Adobe 포스트스크립트 TCP/IP 프린터 간체 한자 Solaris Font Downlodaer
SUNWcj2p	Java 플러그인 1.2.2 간체 한자 버전
SUNWcj2rt	Java 가상 기계와 코어 클래스 라이브러리(간체 한자 추가)
SUNWcjmp	JMF 재생기 간체 한자 버전
SUNWckcsr	간체 한자(EUC) KCMS 런타임 환경
SUNWcleex	간체 한자(EUC) 언어 환경 특정 파일. 간체 한자 언어 환경(64비트) - 확장 실행을 위한 필수 패키지
SUNWcleue	간체 한자(EUC) 언어 환경 특정 파일. 간체 한자(EUC) 언어 환경 실행을 위한 필수 패키지
SUNWclvma	Solaris 볼륨 관리 간체 한자 버전
SUNWclvmg	Solaris 볼륨 관리 간체 한자 버전
SUNWcmga	간체 한자 Solaris 관리 응용프로그램
SUNWcorte	간체 한자(EUC) OPEN LOOK 툴킷 런타임 환경 패키지
SUNWcos	이 패키지는 간체 한자 언어 환경 특정 파일을 포함합니다. 간체 한자 언어 환경 실행을 위한 필수 패키지
SUNWcpdas	데스크탑 응용프로그램을 Palm Pilot PDA와 동기화하는 도구용 간체 한자 버전
SUNWcreg	Solaris 사용자 등록 간체 한자(EUC) 버전

표 C-1 간체 한자 (계속)

패키지 이름	설명
SUNWcrmui	간체 한자 자원 관리 사용자 인터페이스 구성 요소
SUNWcsadl	Solstice Admintool 실행 파일 및 관련 라이브러리 간체 한자(EUC) 버전
SUNWcscgu	간체 한자 버전 Solaris Smart Card 관리 - 그래픽 사용자 인터페이스 구성 요소
SUNWcsfw	SFW 정렬용 간체 한자 버전 메시지 파일
SUNWcsmc	간체 한자 Solaris 관리 콘솔 2.0
SUNWctltk	간체 한자(EUC) ToolTalk 런타임 패키지
SUNWcttfe	간체 한자(EUC) 트루타입 글꼴
SUNWcudc	Solaris CDE 환경용 사용자 정의 문자 도구 간체 한자(EUC) 버전
SUNWcwbc	Solaris WBEM 서비스 간체 한자 버전
SUNWcwbcp	간체 한자(EUC) OpenWindows 이진 호환 패키지
SUNWcwdev	Solaris WBEM 서비스 간체 한자 버전
SUNWcwsr2	Solaris 제품 레지스트리 간체 한자 버전
SUNWcwsrv	Solaris 제품 레지스트리 뷰어 간체 한자 버전
SUNWcxe	간체 한자(EUC) X 윈도우 플랫폼 소프트웨어 패키지
SUNWcxftn	간체 한자(EUC) X 윈도우 플랫폼 필수 글꼴
SUNWcxman	간체 한자(EUC) X 윈도우 온라인 사용 설명서 페이지 패키지
SUNWgttfe	간체 한자(GBK) 트루 타입 글꼴

다음 표는 프랑스어 패키지와 내용을 나열합니다.

표 C-2 프랑스어

패키지 이름	설명
NSCPfrcdo	미국 보안 규정을 지원하는 Netscape Communicator 4.7 프랑스어 버전
NSCPfrcom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 프랑스어 버전
SUNWfbcp	프랑스어 OS 이진 호환 패키지
SUNWfdc1	디스크 없는 클라이언트 관리 응용프로그램 프랑스어 버전
SUNWfdhcm	DHCP Manager 프랑스어 버전
SUNWffdl	Font Downloader를 위한 지역화 가능 문자열
SUNWfj2rt	Java 가상 기계와 코어 클래스 라이브러리(프랑스어 추가)

표 C-2 프랑스어 (계속)

패키지 이름	설명
SUNWfjmf	오디오 및 비디오 파일 재생을 위한 JMF 재생기
SUNWflvma	Solaris 볼륨 관리 API 프랑스어 버전
SUNWflvmg	Solaris 볼륨 관리 응용프로그램 프랑스어 버전
SUNWfmgp	Solaris 관리 응용프로그램 프랑스어 버전
SUNWforte	프랑스어 OPEN LOOK (R) 툴킷 런타임 환경
SUNWfpdas	데스크탑 응용프로그램을 Palm Pilot PDA와 동기화하는 프랑스어 도구
SUNWfrbas	CDE 응용프로그램을 실행하기 위한 기본 L10N 프랑스어 CDE 기능
SUNWfrdis	Admintool과 GUI 설치용 프랑스어 버전
SUNWfrdma	시스템 관리 작업 수행을 위해 사용되는 소프트웨어 프랑스어 버전.
SUNWfrdst	CDE 데스크탑 응용프로그램
SUNWfrdte	CDE 데스크탑 환경
SUNWfrhe	CDE 도움말 L10N 프랑스어 런타임 환경
SUNWfrhed	CDE L10N 프랑스어 도움말 개발자 환경
SUNWfrhev	CDE 도움말 볼륨
SUNWfrhez	Desktop Power Pack 도움말 볼륨 프랑스어 버전
SUNWfrim	CDE 데스크탑 응용프로그램
SUNWfrj2p	Java 플러그인 1.2.2 프랑스어 버전
SUNWfrmui	Solaris용 프랑스어 자원 관리
SUNWfros	OS-네트워킹 정리를 지역화 가능 메시지 파일
SUNWfrpmw	Power Management OW 유틸리티 프랑스어(EUC) 버전
SUNWfrreg	사용자 등록을 위한 데스크탑 로그인시의 Solaris 사용자 등록 프롬프트
SUNWfrsmc	Solaris 관리 콘솔 프랑스어 버전
SUNWfrwbc	Solaris WBEM 서비스 프랑스어 버전
SUNWfrwm	프랑스어 CDE 데스크탑 창 관리자 메시지
SUNWfrws2	프랑스어 버전 Solaris 제품 레지스트리
SUNWfrwsv	Solaris 제품 레지스트리 뷰어용 프랑스어 버전
SUNWfsadl	Solstice Admintool 실행 파일 및 관련 라이브러리 프랑스어 버전
SUNWfscgu	Solaris Smart Card Administration 프랑스어 버전 - 그래픽 사용자 인터페이스 구성 요소

표 C-2 프랑스어 (계속)

패키지 이름	설명
SUNwfsfw	SFW 정리용 프랑스어 버전 메시지 파일
SUNwftltk	프랑스어 ToolTalk 이진 파일과 공유 라이브러리
SUNwfwacx	프랑스어 OPEN LOOK AccessX
SUNwfwbcp	프랑스어 OpenWindows 이진 호환 패키지
SUNwfwdev	프랑스어 Sun WBEM SDK 자원
SUNwfxplt	프랑스어 X 윈도우 플랫폼 소프트웨어

다음 표는 독일어 패키지와 내용을 나열합니다.

표 C-3 독일어

패키지 이름	설명
NSCPdecom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 독일어 버전
SUNWdbcp	독일어 OS 이진 호환 패키지
SUNWddcl	Solaris 디스크 없는 클라이언트 관리 응용프로그램용 독일어 버전
SUNWddhcm	DHCP Manager 독일어 버전
SUNWdebas	CDE 응용프로그램을 실행하기 위한 기본 L10N 독일어 CDE 기능
SUNWdedis	Admintool과 GUI 설치용 독일어 버전
SUNWdedma	시스템 관리 작업 수행을 위해 사용되는 소프트웨어 독일어 버전.
SUNWdedst	CDE 데스크탑 응용프로그램
SUNWdedte	CDE 데스크탑 로그인 환경
SUNWdehe	CDE 도움말 L10N 독일어 런타임 환경
SUNWdehed	CDE L10N 독일어 도움말 개발자 환경
SUNWdehev	CDE 도움말 볼륨
SUNWdehez	Desktop Power Pack 도움말 볼륨 독일어 버전
SUNWdeim	CDE 데스크탑 응용프로그램
SUNWdej2p	Java 플러그인 1.2.2 독일어 버전
SUNWdeos	OS-네트워킹 정리용 지역화 가능 메시지 파일
SUNWdepmw	Power Management OW 유틸리티 독일어(EUC) 버전
SUNWdereg	사용자 등록을 위한 데스크탑 로그인시의 Solaris 사용자 등록 프롬프트

표 C-3 독일어 (계속)

패키지 이름	설명
SUNWdesmc	Solaris 관리 콘솔 독일어 버전
SUNWdewbc	Solaris WBEM 서비스 독일어 버전
SUNWdewm	독일어 CDE 데스크탑 창 관리자 메시지
SUNWdews2	독일어 버전 Solaris 제품 레지스트리
SUNWdewsv	Solaris 제품 레지스트리 뷰어 독일어 버전
SUNWdfdl	Font Downloader를 위한 지역화 가능 문자열
SUNWdj2rt	Java 가상 기계와 코어 클래스 라이브러리(독일어 추가)
SUNWdjmf	오디오 및 비디오 파일 재생을 위한 지역화 가능 JMF 재생기
SUNWdlvma	Solaris 볼륨 관리 API 독일어 버전
SUNWdlvmg	Solaris 볼륨 관리 응용프로그램 독일어 버전
SUNWdmgp	Solaris 관리 응용프로그램 독일어 버전
SUNWdorte	독일어 OPEN LOOK 툴킷 런타임 환경
SUNWdpdas	데스크탑 응용프로그램을 Palm Pilot PDA와 동기화하기 위한 독일어 도구
SUNWdrmui	Solaris용 독일어 자원 관리
SUNWdsadl	Solstice Admintool 실행 파일 및 관련 라이브러리 독일어 버전
SUNWdscgu	지역화 가능 Solaris Smart Card Administration - 그래픽 사용자 인터페이스 구성 요소
SUNWdsfw	SFW 정리용 독일어 버전 메시지 파일
SUNWdtltk	독일어 ToolTalk 이진 파일과 공유 라이브러리
SUNWdwacx	독일어 OPEN LOOK AccessX
SUNWdwbc	독일어 OpenWindows 이진 호환 패키지
SUNWdwdev	독일어 Sun WBEM SDK 자원
SUNWdxplt	독일어 X 윈도우 플랫폼 소프트웨어

다음 표는 이탈리아어 패키지와 내용을 나열합니다.

표 C-4 이탈리아어

패키지 이름	설명
NSCPitcom	국제 보안 규정을 지원하는 Netscape Communicator 4.78 이탈리아어 버전
SUNWibcp	이탈리아어 OS 이진 호환 패키지

표 C-4 이탈리아어 (계속)

패키지 이름	설명
SUNwidcl	디스크 없는 클라이언트 관리 응용프로그램 이탈리아어 지역화
SUNwidhcm	DHCP Manager 이탈리아어 버전
SUNwifdl	Font Downloader를 위한 지역화 가능 문자열
SUNwij2rt	Java 가상 기계와 코어 클래스 라이브러리(이탈리아어 추가)
SUNwijmfp	오디오 및 비디오 파일 재생을 위한 지역화 가능 JMF 재생기
SUNwilvma	Solaris 볼륨 관리 API 이탈리아어 버전
SUNwilvmg	Solaris 볼륨 관리 응용프로그램 이탈리아어 버전
SUNwimgp	Solaris 관리 응용프로그램 이탈리아어 버전
SUNwiorte	이탈리아어 OPEN LOOK (R) 툴킷 런타임 환경
SUNwipdas	데스크탑 응용프로그램을 Palm Pilot PDA와 동기화하는 이탈리아어 도구
SUNwirmui	Solaris용 이탈리아어 자원 관리
SUNwisadl	Solstice Admintool 실행 파일 및 관련 라이브러리 이탈리아어 버전
SUNwiscpu	지역화 가능 Solaris Smart Card Administration - 그래픽 사용자 인터페이스 구성 요소
SUNwisfw	SFW 정리용 이탈리아어 버전 메시지 파일
SUNwitbas	CDE 응용프로그램을 실행하기 위한 표준 L10N 이탈리아어 CDE 기능
SUNwitdis	Admintool과 GUI 설치용 이탈리아어 버전
SUNwitdma	시스템 관리 작업 수행을 위해 사용되는 소프트웨어 이탈리아어 버전.
SUNwitdst	CDE 이탈리아어 데스크탑 응용프로그램 메시지
SUNwitdte	CDE 이탈리아어 데스크탑 로그인 환경
SUNwithe	CDE 도움말 L10N 이탈리아어 런타임 환경
SUNwithed	CDE L10N 이탈리아어 도움말 개발자 환경
SUNwithhev	CDE 도움말 볼륨
SUNwithez	CDE 도움말 볼륨 이탈리아어 버전
SUNwitim	CDE 이탈리아어 데스크탑 이미지 편집기
SUNwitj2p	Java 플러그인 1.2.2 이탈리아어 버전
SUNwitltk	이탈리아어 ToolTalk 이진 파일과 공유 라이브러리
SUNwitos	OS-네트워킹 정리용 지역화 가능 메시지 파일
SUNwitpmw	Power Management OW 유틸리티 이탈리아어(EUC) 버전

표 C-4 이탈리아어 (계속)

패키지 이름	설명
SUNWitreg	사용자 등록을 위한 데스크탑 로그인시의 Solaris 사용자 등록 프롬프트
SUNWitsmc	Solaris 관리 콘솔 이탈리아어 버전
SUNWitwbc	Solaris WBEM 서비스 이탈리아어 버전
SUNWitwm	이탈리아어 CDE 데스크탑 창 관리자 메시지
SUNWitws2	이탈리아어 버전 Solaris 제품 레지스트리
SUNWitwsv	Solaris 제품 레지스트리 뷰어 이탈리아어 버전
SUNWiwacx	이탈리아어 OPEN LOOK AccessX
SUNWiwbcp	이탈리아어 OpenWindows 이진 호환 패키지
SUNWiwdev	이탈리아어 Sun WBEM SDK 자원
SUNWixplt	이탈리아어 X 윈도우 플랫폼 소프트웨어

다음 표는 일본어 패키지와 내용을 나열합니다.

표 C-5 일본어

패키지 이름	설명
IPLTjadcon	관리 서버 콘솔 클라이언트 패키지 일본어 버전
IPLTjadman	관리 서버 설명서 일본어 버전
IPLTjadmin	관리 서버 일본어 버전
IPLTjcons	콘솔 클라이언트 기반 패키지 일본어 버전
IPLTjdscon	디렉토리 서버 콘솔 일본어 버전
IPLTjdsman	디렉토리 서버 설명서 일본어 버전
IPLTjdsu	디렉토리 서버 일본어 버전
NSCPjecom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 일본어(EUC) 버전
NSCPjpcom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 일본어(PCK) 버전
NSCPjucom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 일본어(UTF-8) 버전
SUNWjadcl	Solaris 디스크 없는 클라이언트 관리 응용프로그램용 일본어 버전
SUNWjadis	Admintool과 GUI 설치용 일본어(EUC) 버전
SUNWjadma	시스템 관리 작업 수행을 위해 사용되는 소프트웨어 일본어(EUC) 버전. Admintool은 이 패키지와 일본어(EUC) 버전용 SUNWjadis 패키지를 필요로 합니다.
SUNWjaj2p	Java 플러그인 1.2.2 일본어 버전



표 C-5 일본어 (계속)

패키지 이름	설명
SUNWjbcp	SunOS 4.x 응용프로그램을 위한 이진 호환 실행 환경을 제공하는 libc와 로캘 데이터를 포함한 일본어(EUC) 유틸리티
SUNWjcs3f	일본어 JIS X0212 유형1 인쇄 글꼴
SUNWjdab	CDE Desktop Application Builder 일본어(공통) 버전
SUNWjdbas	CDE 응용프로그램 기본 런타임 환경 일본어(공통) 버전
SUNWjddst	CDE 데스크탑 응용프로그램 일본어(EUC) 버전
SUNWjddte	Solaris 데스크탑 로그인 환경 일본어(EUC) 버전
SUNWjdhcm	DHCP Manager 일본어 버전
SUNWjdhe	CDE 도움말 런타임 환경 일본어(EUC) 버전
SUNWjdhed	CDE 도움말 개발자 환경 일본어(EUC) 버전
SUNWjdhev	CDE 도움말 볼륨 일본어(공통) 버전
SUNWjdhez	Desktop Power Pack 도움말 볼륨 일본어(공통) 버전
SUNWjdim	Solaris CDE Image Viewer 일본어(EUC) 버전
SUNWjdma	CDE MAN PAGES 일본어 버전
SUNWjdoc	일본어 설명서 도구
SUNWjdwmm	CDE 데스크탑 창 관리자 일본어(EUC) 버전
SUNWject	UTF-8 코드 변환 도구 일본어(EUC) 버전
SUNWjedev	일본어(EUC) 개발 환경 패키지 특정 파일
SUNWjeezt	Desktop Power Pack Applications 일본어(EUC) 버전
SUNWjej2m	일본어(EUC) 설명서 페이지
SUNWjeman	SUNWjfr과 SUNWjfp우용 일본어(EUC) 설명서 페이지와 SUNWman과 SUNWaled용 일본어 설명서 페이지를 보기 위한 일본어 기능 패키지 설명서 페이지
SUNWjepmm	일본어(EUC) Power Management OW 유틸리티 설명서 페이지
SUNWjepmw	Power Management OW 유틸리티 일본어(EUC) 버전
SUNWjeudc	Solaris CDE 환경용 사용자 정의 문자 도구 일본어(EUC) 버전
SUNWjfdl	Adobe PostScript 프린터용 Solaris Desktop Font Downloader 일본어 버전
SUNWjfpref	일본어 기능 패키지(JFP)용 스트림 모듈. 이 파일들은 JFP 환경을 실행하기 위한 확장 패키지임.
SUNWjfpue	사용자용 일본어 기능 패키지(JFP) 특정 파일. 이 파일들은 JFP 환경을 실행하기 위한 확장 패키지임.

표 C-5 일본어 (계속)

패키지 이름	설명
SUNWjfxmn	X 윈도우 시스템용 일본어 기능에 대한 영어 설명서 페이지
SUNWjj2rt	일본어 Java 가상 기계와 코어 클래스 라이브러리
SUNWjjmfp	JMF 재생기 일본어 버전
SUNWjkcsr	Kodak Color Management System 런타임 일본어(EUC) 버전
SUNWjlvma	Solaris 볼륨 관리 API 일본어 버전
SUNWjlvmg	Solaris 볼륨 관리 응용프로그램 일본어 버전
SUNWjmane	SUNWjfpref와 SUNWjfpue에 관한 영어 설명서 페이지를 보기 위한 일본어 기능 패키지 설명서 페이지(확장)
SUNWjmfrn	Motif 1.2.3 런타임 키트 일본어(EUC) 버전
SUNWjmga	일본어 Solaris 관리 응용프로그램
SUNWjorte	OPEN LOOK 툴킷 런타임 환경 일본어(EUC) 버전
SUNWjos	일본어 ON 메시지 파일
SUNWjpdas	데스크탑 응용프로그램을 Palm Pilot PDA와 동기화하는 도구 일본어 버전
SUNWjpp2m	일본어(PCK) 설명서 페이지
SUNWjreg	Solaris 사용자 등록 일본어 버전
SUNWjrmui	Solaris용 일본어 자원 관리
SUNWjsadl	Solstice Admintool 실행 파일 및 관련 라이브러리 일본어(EUC) 버전
SUNWjscag	Solaris Smart Card Administration 일본어 버전 - 그래픽 사용자 인터페이스 구성 요소
SUNWjsfw	SFW 정리용 일본어 지역화 가능 메시지 파일
SUNWjsmc	일본어 Solaris Management Console 2.0
SUNWjt1mn	ToolTalk 프로그래머, OpenWindows 사용자 및 공통 데스크탑 환경(CDE) 사용자용 일본어(EUC) ToolTalk 설명서 페이지
SUNWjt1tk	공통 데스크탑 환경(CDE), OpenWindows 및 모든 ToolTalk 클라이언트에 필요한 ToolTalk 이진 파일과 공유 라이브러리 일본어(EUC) 버전
SUNWjuj2m	일본어(UTF-8) 설명서 페이지
SUNWjwacx	AccessX 클라이언트 프로그램 일본어(EUC) 버전
SUNWjwbc	Solaris WBEM 서비스 일본어 버전
SUNWjwbcp	Openwindows 이진 호환용 지원 파일, 프로그램 및 라이브러리 일본어(EUC) 버전
SUNWjwbd	Sun WBEM SFW 자원 일본어 버전

표 C-5 일본어 (계속)

패키지 이름	설명
SUNWjwncr	일본어 입력 시스템, Wnn6 클라이언트, (루트)
SUNWjwncu	일본어 입력 시스템 - Wnn6 클라이언트, (Usr)
SUNWjwncx	일본어 입력 시스템 - Wnn6 클라이언트 X 윈도우 시스템
SUNWjwndt	일본어 입력 시스템 - CDE용 Wnn6 클라이언트
SUNWjwnsr	일본어 입력 시스템 - Wnn6 클라이언트, (루트)
SUNWjwnsu	일본어 입력 시스템 - Wnn6 서버, (Usr)
SUNWjws2	Solaris 제품 레지스트리 일본어 버전
SUNWjwsv	Solaris 제품 레지스트리 뷰어 일본어 버전
SUNWjxfa	Solaris 플랫폼용 글꼴 관리 응용프로그램 일본어(공통) 버전
SUNWjxfnt	일본어 X 윈도우 시스템 글꼴(필수 글꼴) - 고딕 굵은체 글꼴과 트루타입 맵 파일
SUNWjxim	일본어 X 윈도우 시스템 X 입력 방식 서버 패키지
SUNWjxoft	Sun Minchou 비트맵 글꼴
SUNWjxpmn	일본어(EUC) X 윈도우 시스템 온라인 프로그래머 설명서 페이지
SUNWjxumn	일본어(EUC) X 윈도우 시스템 온라인 사용 설명서 페이지

다음 표는 한국어 패키지와 내용을 나열합니다.

표 C-6 한국어

패키지 이름	설명
NSCPkocom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 한국어 버전
NSCPkucom	국제 보안 규정을 지원하는 Netscape Communicator 4.78 ko.UTF-8 버전
SUNWkacx	한국어 AccessX 클라이언트 프로그램
SUNWkadis	Admintool과 GUI 설치용 한국어 버전
SUNWkadma	시스템 관리 작업 수행을 위해 사용되는 소프트웨어 한국어 버전. Admintool은 이 패키지와 한국어(EUC) 버전용 SUNWjadis 패키지를 필요로 합니다.
SUNWkbcp	이 패키지는 한국어 환경 이진 호환 파일을 포함합니다.
SUNWkcoft	한국어/한국어 UTF-8 공통 선택 글꼴 패키지
SUNWkdab	CDE Desktop Application Builder 한국어 버전
SUNWkdbas	CDE Base 기능 한국어 버전
SUNWkdcl	Solaris 디스크 없는 클라이언트 관리 응용프로그램용 한국어 버전

표 C-6 한국어 (계속)

패키지 이름	설명
SUNWkdcst	한글화된 도구 패키지
SUNWkddst	CDE 데스크탑 응용프로그램 한국어 버전
SUNWkddte	CDE 데스크탑 로그인 환경 한국어 버전
SUNWkdez	Desktop Power Pack Applications 한국어(EUC) 버전
SUNWkdft	공통 데스크탑 환경용 글꼴, 한국어 L10N CDE
SUNWkdhcm	DHCP Manager 한국어 버전
SUNWkdhe	CDE 도움말 런타임 환경 한국어 버전
SUNWkdhev	한국어 CDE 도움말 볼륨
SUNWkdhez	Desktop Power Pack 도움말 볼륨 한국어(공통) 버전
SUNWkdicn	CDE 아이콘 한국어 버전
SUNWkdim	CDE Imagetool 한국어 버전
SUNWkdwm	CDE 데스크탑 창 관리자 한국어 버전
SUNWkepmw	Power Management OW 유틸리티 한국어(EUC) 버전
SUNWkfdl	Adobe 포스트스크립트 TCP/IP 프린터 한국어 Solaris Font Downloader
SUNWkj2rt	Java 가상 기계와 코어 클래스 라이브러리(한국어 추가)
SUNWkjmfj	JMF 재생기 한국어 버전
SUNWkkcsr	한국어(EUC) KCMS 런타임 환경
SUNWkleex	한국어 언어 환경 특정 파일. 한국어 언어 환경(64비트) 실행을 위한 필수 패키지
SUNWkler	이 패키지는 한국어 환경용 스트림 모듈을 포함합니다. 한국어 언어 환경 실행을 위한 필수 패키지
SUNWkleue	이 패키지는 한국어 환경 이진 특정 파일을 포함합니다. 한국어 언어 환경 실행을 위한 필수 패키지
SUNWklvma	Solaris 볼륨 관리 API 한국어 버전
SUNWklvmg	Solaris 볼륨 관리 응용프로그램 한국어 버전
SUNWkmga	한국어 Solaris 관리 응용프로그램
SUNWkoj2p	Java 플러그인 1.2.2 한국어 버전
SUNWkorte	한국어 OPEN LOOK 툴킷 런타임 환경 패키지
SUNWkos	이 패키지는 한국어 환경 이진 특정 파일을 포함합니다. 한국어 언어 환경 실행을 위한 필수 패키지
SUNWkpdas	데스크탑 응용프로그램을 Palm Pilot PDA와 동기화하는 도구용 한국어 버전

표 C-6 한국어 (계속)

패키지 이름	설명
SUNWkreg	Solaris 사용자 등록 한국어 버전
SUNWkrmui	한국어 자원 관리 사용자 인터페이스 구성 요소
SUNWksadl	Solstice Admintool 실행 파일 및 관련 라이브러리 한국어 버전
SUNWkscgu	Solaris Smart Card Administration 한국어 버전 - 그래픽 사용자 인터페이스 구성 요소
SUNWksfw	SFW 정리용 한국어 지역화 가능 메시지 파일
SUNWksmc	한국어 Solaris 관리 콘솔 2.0
SUNWkttltk	한국어 ToolTalk 런타임 패키지
SUNWkttfe	한국어 트루 타입 글꼴 확장
SUNWkudc	Solaris CDE 환경용 사용자 정의 문자 도구 한국어(EUC) 버전
SUNWkuxft	한국어 UTF-8 X 윈도우 시스템 플랫폼 필수 글꼴
SUNWkwbc	Solaris WBEM 서비스 한국어 버전
SUNWkwbcp	한국어 OpenWindows 이진 호환 패키지
SUNWkwdev	Solaris WBEM 서비스 한국어 버전
SUNWkwsr	한국어 prodreg 2.0 지역화 가능 텍스트 자원
SUNWkwsr2	Solaris 제품 레지스트리 한국어 버전
SUNWkwsrv	Solaris 제품 레지스트리 뷰어 한국어 버전
SUNWkxe	한국어 X 윈도우 시스템 플랫폼 소프트웨어 패키지
SUNWkxfte	한국어 X 윈도우 시스템 플랫폼 필수 글꼴
SUNWkxman	한국어 X 윈도우 시스템 온라인 사용 설명서 페이지 패키지

다음 표는 스페인어 패키지와 내용을 나열합니다.

표 C-7 스페인어

패키지 이름	설명
NSCPescom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 스페인어 버전
SUNWcudc	디스크 없는 클라이언트 관리 응용프로그램 스페인어 버전
SUNWedhcm	DHCP Manager 스페인어 버전
SUNWefdl	Font Downloader를 위한 지역화 가능 문자열
SUNWej2rt	Java 가상 기계와 코어 클래스 라이브러리(스페인어 추가)

표 C-7 스페인어 (계속)

패키지 이름	설명
SUNWejmfsp	오디오 및 비디오 파일 재생을 위한 지역화 가능 JMF 재생기
SUNWelvma	Solaris 볼륨 관리 API 스페인어 버전
SUNWelvmg	Solaris 볼륨 관리 응용프로그램 스페인어 버전
SUNWemgp	Solaris 관리 응용프로그램 스페인어 버전
SUNWeorte	스페인어 OPEN LOOK (R) 툴킷 런타임 환경
SUNWepdas	데스크탑 응용프로그램을 Palm Pilot PDA와 동기화하는 도구 스페인어 버전
SUNWermui	Solaris용 스페인어 자원 관리
SUNWesadl	Solstice Admintool 실행 파일 및 관련 라이브러리 스페인어 버전
SUNWesbas	CDE 응용프로그램을 실행하기 위한 기본 L10N 프랑스어 CDE 기능
SUNWescgu	지역화 가능 Solaris Smart Card 관리 - 그래픽 사용자 인터페이스 구성 요소
SUNWesdis	Admintool과 GUI 설치용 스페인어 버전
SUNWesdma	시스템 관리 작업 수행을 위해 사용되는 소프트웨어 스페인어 버전.
SUNWesdst	CDE 데스크탑 응용프로그램
SUNWesdte	CDE 데스크탑 로그인 환경
SUNWesfw	SFW 정리용 스페인어 지역화 가능 메시지 파일
SUNWeshe	CDE 도움말 L10N 스페인어 런타임 환경
SUNWeshed	CDE L10N 스페인어 도움말 개발자 환경
SUNWeshev	CDE 도움말 볼륨
SUNWeshez	Desktop Power Pack 도움말 볼륨 스페인어 버전
SUNWesim	CDE 데스크탑 응용프로그램
SUNWesj2p	Java 플러그인 1.2.2 스페인어 버전
SUNWesos	OS-네트워킹 정리용 지역화 가능 메시지 파일
SUNWespmw	Power Management OW 유틸리티 스페인어(EUC) 버전
SUNWesreg	사용자 등록을 위한 데스크탑 로그인시의 Solaris 사용자 등록 프롬프트
SUNWessmc	Solaris 관리 콘솔 스페인어 버전
SUNWeswbc	Solaris WBEM 서비스 스페인어 버전
SUNWeswm	스페인어 CDE 데스크탑 창 관리자 메시지
SUNWesws2	스페인어 버전 Solaris 제품 레지스트리

표 C-7 스페인어 (계속)

패키지 이름	설명
SUNWeswsv	Solaris 제품 레지스트리 뷰어 스페인어 버전
SUNWetltk	스페인어 ToolTalk 이진 파일과 공유 라이브러리
SUNWewacx	스페인어 OPEN LOOK AccessX
SUNWewdev	스페인어 Sun WBEM SDK 자원
SUNWexplt	스페인어 X 윈도우 플랫폼 소프트웨어

다음 표는 스웨덴어 패키지와 내용을 나열합니다.

표 C-8 스웨덴어

패키지 이름	설명
NSCPsvcom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 스웨덴어 버전
SUNWsdcl	디스크 없는 클라이언트 관리 응용프로그램 스웨덴어 버전
SUNWsdhcm	DHCP Manager 스웨덴어 버전
SUNWsfdl	Font Downloader를 위한 지역화 가능 문자열
SUNWsj2rt	Java 가상 기계와 코어 클래스 라이브러리(스웨덴어 추가)
SUNWsjmfp	오디오 및 비디오 파일 재생을 위한 지역화 가능 JMF 재생기
SUNWslvma	Solaris 볼륨 관리 API 스웨덴어 버전
SUNWslvmg	Solaris 볼륨 관리 응용프로그램 스웨덴어 버전
SUNWsmgp	Solaris 관리 응용프로그램 스웨덴어 버전
SUNWsorte	스웨덴어 OPEN LOOK (R) 툴킷 런타임 환경
SUNWspdas	데스크탑 응용프로그램을 Palm Pilot PDA와 동기화하기 위한 스웨덴어 도구
SUNWsrmui	Solaris용 스웨덴어 자원 관리
SUNWssadl	Solstice Admintool 실행 파일 및 관련 라이브러리 스웨덴어 버전
SUNWsscgu	지역화 가능 Solaris Smart Card Administration - 그래픽 사용자 인터페이스 구성 요소
SUNWssfsw	SFW 정리용 스웨덴어 지역화 가능 메시지 파일
SUNWstltk	스웨덴어 ToolTalk 이진 파일과 공유 라이브러리
SUNWsvbas	표준 스웨덴어 CDE 기능 메시지
SUNWsvdis	Admintool과 GUI 설치용 스웨덴어 버전
SUNWsvdma	시스템 관리 작업 수행을 위해 사용되는 소프트웨어 스웨덴어 버전.

표 C-8 스웨덴어 (계속)

패키지 이름	설명
SUNWsvdst	스웨덴어 CDE 데스크탑 응용프로그램 메시지
SUNWsvdte	스웨덴어 CDE 데스크탑 로그인 환경 메시지
SUNWsvhe	스웨덴어 CDE 도움말 런타임 환경
SUNWsvhed	스웨덴어 CDE 도움말 개발자 환경 메시지
SUNWsvhev	CDE 도움말 블록
SUNWsvhez	Desktop Power Pack 도움말 블록 스웨덴어 버전
SUNWsvim	스웨덴어 CDE 이미지 편집기 메시지
SUNWsvj2p	Java 플러그인 1.2.2 스웨덴어 버전
SUNWsvos	OS-네트워킹 정리용 지역화 가능 메시지 파일
SUNWsvpmw	Solaris 블록 관리 Open Windows 스웨덴어(EUC) 버전유틸리티
SUNWsvreg	사용자 등록을 위한 데스크탑 로그인시의 Solaris 사용자 등록 프롬프트
SUNWsvsmc	Solaris 관리 콘솔 스웨덴어 버전
SUNWsvwbc	Solaris WBEM 서비스 스웨덴어 버전
SUNWsvwm	스웨덴어 CDE 데스크탑 창 관리자 메시지
SUNWsvws2	스웨덴어 버전 Solaris 제품 레지스트리
SUNWsvwsv	Solaris 제품 레지스트리 뷰어 스웨덴어 버전
SUNWsvwsv	Solaris 제품 레지스트리 뷰어 스웨덴어 버전
SUNWswacx	스웨덴어 OPEN LOOK AccessX
SUNWsxplt	스웨덴어 X 윈도우 플랫폼 소프트웨어
SUNWvbcp	스웨덴어 OS 이진 파일 호환 패키지

다음 표는 정체 한자 패키지와 내용을 나열합니다.

표 C-9 정체 한자

패키지 이름	설명
NSCP5com	국제 보안 규정을 지원하는 Netscape Communicator 4.7 Zh_TW.BIG5 버전
NSCPcom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 정체 한자 버전
NSCPucom	국제 보안 규정을 지원하는 Netscape Communicator 4.7 Zh_TW.UTF-8 버전
SUNW5tttfe	정체 한자 트루 타입 글꼴 패키지 확장



표 C-9 정체 한자 (계속)

패키지 이름	설명
SUNW5xfnt	정체 한자BIG5 X 윈도우 시스템 플랫폼 필수 글꼴 패키지
SUNWhacx	정체 한자 AccessX 클라이언트 프로그램.
SUNWhadis	Admintool과 GUI 설치용 정체 한자(EUC) 버전
SUNWhadma	시스템 관리 작업 수행을 위해 사용되는 소프트웨어 정체 한자(EUC) 버전. Admintool은 이 패키지와 정체 한자(EUC) 버전용 SUNWhadis 패키지를 필요로 합니다.
SUNWhbcp	이 패키지는 정체 한자 언어 환경 이진 호환 파일을 포함합니다.
SUNWhdab	CDE Desktop Application Builder 정체 한자 버전
SUNWhdbas	CDE Base 기능 정체 한자 버전
SUNWhdcl	Solaris 디스크 없는 클라이언트 관리 응용프로그램을 위한 정체 한자 버전
SUNWhddst	CDE 데스크탑 응용프로그램 정체 한자 버전
SUNWhddte	CDE 데스크탑 로그인 환경 정체 한자 버전
SUNWhdez	Desktop Power Pack Applications 정체 한자(EUC) 버전
SUNWhdft	CDE 글꼴 정체 한자 버전
SUNWhdhcm	DHCP Manager 정체 한자 버전
SUNWhdhe	CDE 도움말 런타임 환경 정체 한자 버전
SUNWhdhev	정체 한자 CDE 도움말 볼륨
SUNWhdhez	Desktop Power Pack 도움말 볼륨 정체 한자(공통) 버전
SUNWhdicn	CDE 아이콘 정체 한자 버전
SUNWhdim	CDE Imagetool 정체 한자 버전
SUNWhdwm	CDE 데스크탑 창 관리자 정체 한자 버전
SUNWhepmw	Power Management OW 유틸리티 정체 한자(EUC) 버전
SUNWhfdl	Adobe 포스트스크립트 TCP/IP 프린터 정체 한자 Solaris Font Downloader
SUNWhj2p	Java 플러그인 1.2.2 정체 한자 버전
SUNWhj2rt	Java 가상 기계와 코어 클래스 라이브러리(정체 한자 추가)
SUNWhjmfp	JMF 재생기 정체 한자 버전
SUNWhkcsr	정체 한자(EUC) KCMS 런타임 환경
SUNWhkdc1	Solaris 디스크 없는 클라이언트 관리 응용프로그램 정체 한자(홍콩) 버전

표 C-9 정체 한자 (계속)

패키지 이름	설명
SUNWhkeex	정체 한자(홍콩 BIG5) 언어 환경 특정 파일이것은 정체 한자(홍콩 BIG5) 언어 환경 (64비트) 실행을 위한 필수 패키지
SUNWhkeue	이 패키지는 정체 한자(홍콩) 언어 환경 특정 파일을 포함합니다.이 패키지는 정체 한자(홍콩) 언어 환경 실행을 위한 필수 패키지입니다.
SUNWhklvmg	Solaris 디스크 없는 클라이언트 관리 응용프로그램 정체 한자(홍콩) 버전
SUNWhkmg	정체 한자(홍콩) Solaris 관리 응용프로그램
SUNWhkrmui	정체 한자(홍콩) 자원 관리 사용자 인터페이스 구성 요소
SUNWhksmc	정체 한자(홍콩) Solaris 관리 콘솔 2.0
SUNWhkxe	정체 한자(홍콩) X 윈도우 플랫폼 소프트웨어 패키지
SUNWhleex	정체 한자 언어 환경 특정 파일정체 한자 언어 환경(64비트) 실행을 위한 필수 패키지
SUNWhleue	정체 한자 언어 환경 특정 파일정체 한자 언어 환경 실행을 위한 필수 패키지
SUNWhlvma	Solaris 볼륨 관리 API 정체 한자 버전
SUNWhlvmg	Solaris 볼륨 관리 응용프로그램용 정체 한자 버전
SUNWhmga	Solaris 관리 응용프로그램 정체 한자 버전
SUNWhorte	정체 한자 OPEN LOOK 툴킷 런타임 환경 패키지
SUNWhos	이 패키지는 정체 한자 언어 환경 특정 파일을 포함합니다.정체 한자 언어 환경 실행을 위한 필수 패키지
SUNWhpdas	데스크탑 응용프로그램과 Palm Pilot PDA를 동기화하는 도구 정체 한자 버전
SUNWhreg	Solaris 사용자 등록 정체 한자 버전
SUNWhrmui	정체 한자 자원 관리 사용자 인터페이스 구성 요소
SUNWhsadl	Solstice Admintool 실행 파일 및 관련 라이브러리 정체 한자 버전
SUNWhscgu	정체 한자 지역화 가능 Solaris Smart Card Administration - 그래픽 사용자 인터페이스 구성 요소
SUNWhsfw	SFW 통합용 정체 한자 지역화 가능 메시지 파일
SUNWhsmc	정체 한자 Solaris 관리 콘솔 2.0
SUNWhtltk	정체 한자 ToolTalk 런타임 패키지
SUNWhttfe	정체 한자 트루타임 선택 글꼴 패키지 확장
SUNWhudc	Solaris CDE 환경용 사용자 정의 문자 도구 정체 한자(EUC) 버전
SUNWhwbc	Solaris WBEM 서비스 정체 한자 버전
SUNWhwbcp	정체 한자 OpenWindows 이진 파일 호환 패키지

표 C-9 정체 한자 (계속)

패키지 이름	설명
SUNWhwdev	Solaris WBEM 서비스 정체 한자 버전
SUNWhwsr2	Solaris 제품 레지스트리 정체 한자 버전
SUNWhwsrv	Solaris 제품 레지스트리 뷰어 정체 한자 버전
SUNWhxe	정체 한자 X 윈도우 플랫폼 소프트웨어 패키지
SUNWhxman	정체 한자 X 윈도우 온라인 사용 설명서 페이지 패키지

다음 표는 공유 패키지와 내용을 나열합니다.

표 C-10 아시아권 공유

패키지 이름	설명
SUNWabcp	SunOS 4.x 이진 호환을 위한 아시아권 공통 파일
SUNWudct	Solaris CDE 환경용 사용자 정의 문자 도구



# 색인

---

## 번호와 기호

@ls numerals=nominal:national, 126  
16비트 유니코드 3.0 코드 세트, 135

## A

API, 43  
AttrObject, 125

## C

catgets(), 50  
CDE, en\_US.UTF-8 로케일 지원, 92  
CDE 입력 방식, 94  
Complex Text Layout (CTL)  
  Motif, 123  
  Motif 라이브러리 이식 지침, 141  
  rendition 만들기, 135  
  rendition 편집하기, 135  
  레이아웃 방향, 133  
  렌더 테이블 만들기, 136  
  마우스 선택, 139  
  수평 탭, 138  
  키보드 선택, 140  
  텍스트 자원 및 기하학, 140  
Complex Text Layout(CTL)  
  PLS, 121  
  X 라이브러리 출력 모듈(XOM), 122  
  XOC 자원, 122  
  구별 기호, 121  
  구조 개요, 122

## Complex Text Layout(CTL) (계속)

기술 개요, 121  
동적 텍스트 위젯트, 121  
문자 모양, 121  
복잡한 언어 모양, 121  
세그먼트 정렬, 121  
슬러, 121  
언어 엔진, 121  
정적 및 동적 텍스트, 121  
탭, 121  
텍스트 렌더링, 121  
텍스트 방향, 121

## Compose 키, 35

Context, 126  
.cshrc, 116  
CSText, 126  
CTL(Complex Text Layout), 16  
ctype 매크로, 42

## D

DST (일광 절약 시간), 28  
Dt Apps, 122  
DtMail, MIME 문자 세트, 117  
dtterm, 114

## E

editing behavior, 126  
en\_US.UTF-8  
  fontset 정의, 120

en\_US.UTF-8 (계속)  
글꼴 세트 정의, 120  
지원, 92

## F

FontSet 정의, 120

## G

genmsg 유틸리티, 50  
GMT 오프셋, 28

## H

HKSCS iconv 모듈, 22

## I

IA, 키보드, 74  
iconv, 일본어 문자 코드 변환, 86  
iconv 변환 모듈  
  IBM/Microsoft EBCDIC, 180  
  IBM/Microsoft EBCDIC/PC 코드 페이지, 182  
  PC 코드 페이지, 180  
  유니코드 관련, 159  
ISO-10646, 92  
ISO Latin-1, 25  
ISO8859-n 문자 지원, 93

## L

LANG, 환경 변수, 113  
LANG 환경 변수, 112  
layout behavior, 126  
layoutDirection, 133  
LayoutObject, 124, 125  
LC\_ALL, 25  
libc, 41, 43  
libc  
  날짜 및 시간 형식, 47  
  넓은 문자, 44

## libc (계속)

  넓은 문자 및 문자열 처리, 47  
  넓은 문자 입출력, 49  
  넓은 문자열, 49  
  로컬 데이터 질의, 45  
  로컬 수정 및 질의, 44  
  멀티바이트 처리, 47  
  메시지 처리, 43  
  문자 분류 및 지역, 45  
  문자 조합, 46  
  정규 표현식, 44  
  코드 변환, 44  
  통화 처리, 46  
  형식화된 넓은 문자 입출력, 48

locale utility, 112

## M

m\_create\_layout(), 126  
mbtowc(), 43  
modinfo command, 114  
mp, 22, 143, 144  
mp, 16  
mystreams 파일, 116

## N

Numerals, 135  
NUMERALS\_CONTEXTUAL, 135  
NUMERALS\_NATIONAL, 126  
NUMERALS\_NOMINAL, 126

## O

Orientation, 126  
OutToInp, 127

## P

Property, 127  
pseudo-XOC, 124

## S

/sbin/sh command, 41  
setlocale 설명서 페이지, 113  
ShapeCharset, 126, 135  
Solaris 9  
  기능, 22  
  로캘 콘텐츠, 55, 75  
  아시아권 로캘, 75  
  일본어 입력 시스템, 85  
  중국어 문자 세트, 80  
  한국어 지역화, 87  
SPARC 키보드, 66  
strchg command, 115  
strconf command, 115  
STREAMS 모듈  
  32비트 STREAMS, 113  
  64비트 STREAMS, 113  
  TTY 환경 설정, 113  
  u8lat1, 115  
  u8lat2, 115  
  로드, 114  
  설정 저장, 116

## T

TextField, 133  
TextShaping, 126  
TTY 환경 설정, 113  
Tuthill, Bill, 18  
TypeOfText, 126

## U

UIL, 133  
Unicode 3.1, 지원, 92  
Unicode의 개요, 91  
/usr/include/ctype.h, ctype 매크로, 42  
/usr/ucb/stty, use /bin/stty 디렉토리, 115  
UTC, 28  
UTF-8 encoding, 92  
utility, locale, 112

## X

X/Open-Uniform Joint Internationalization  
  Working Group, 92  
X 논리적 글꼴 설명, 148  
XLFD, 148  
XmALIGNMENT\_CENTER, 127  
XmALIGNMENT\_END, 127  
XmCR\_MOVING\_INSERT\_CURSOR, 128  
XmDEFAULT\_DIRECTION, 124  
XmEDIT\_LOGICAL, 127, 129, 139  
XmEDIT\_VISUAL, 127, 139  
XmFont\_IS\_XO, 125  
XmFONT\_IS\_XOC, 125, 137  
XmLabel, 124, 135  
XmLabelG, 124  
XmList, 124  
XmNalignment, 127, 133  
XmNAlignment, 134  
XmNeditPolicy, 127, 139  
  left-character(), 128  
XmNfont, 125  
XmNfontName, 125  
XmNfontType, 125  
XmNgainPrimaryCallback, 128  
XmNlabelString, 135  
XmNlayoutAttrObject, 125  
XmNlayoutDirection, 124, 125, 132, 133  
XmNlayoutModifier, 124, 125, 126, 133, 134  
XmNlayoutModifier, 126  
XmNmotionVerifyCallback, 128  
XmNrenderTable, 127, 140  
XmNrenditionTag, 127  
XmRenderTableAddRenditions, 137  
XmRendition, 123, 124, 126, 127, 133  
XmRendition{Retrieve,Update}, 126  
XmString, 125, 132  
XmSTRING\_COMPONENT\_LAYOUT\_PUSH, 124  
XmSTRING\_COMPONENT\_LOCALE\_TEXT, 124  
XmSTRING\_COMPONENT\_TEXT, 124  
XmSTRING\_COMPONENT\_WIDECHAR\_TEXT, 124  
XmStringDirection, 125, 132  
XmStringDirectionCreate, 132  
XmText, 124, 126, 133  
XmTextField, 124, 126  
XmTextFieldGetLayoutModifier, 131  
XmTextFieldSetLayoutModifier, 131  
XmTextGetLayoutModifier, 131  
XmTextSetLayoutModifier, 132

XoJIG, 92  
XPG4 응용 프로그램, 42  
xpr, 151

## 간

간체 및 정체 한자  
  보조 창 지원, 77  
  입력 방식, 76  
간체 한자  
  locales, 78  
  문자 지원, 93  
  입력 방식, 79  
간체 한자 입력 모드, 110  
간체 한자 패키지, 193

## 개

개발자의 클러스터, 93

## 공

공통 프롤로그 파일, 151

## 구

구성 파일의 지역화, 145

## 국

국제화  
  ISO Latin-1, 25  
  개념, 23  
국제화 API, 43

## 그

그리니치 표준시로부터의 오프셋, 28  
그리스어  
  문자 지원, 93  
  입력 모드, 106, 110  
그리스어 UNIX 키보드, 107

그리스어 유럽 키보드, 107  
그리스어 입력 모드, 106

## 글

글꼴 그룹 정의, 146  
글꼴 그룹에 매핑, 146  
글꼴 별명 짓기, 146  
글꼴 세트 정의, 120

## 날

날짜 형식, 28

## 넓

넓은 문자, 프로세스 코드, 43  
넓은 문자 표현식, 37

## 단

단말기  
  라틴-n, 115  
  라틴-n 단말기, 115  
단말기 옵션, 설정, 115

## 독

독일어, 문자 지원, 93  
독일어 패키지, 197

## 동

동적 연결, 41

## 라

라이브러리, 응용프로그램 연결, 41  
라틴-1, 라틴-2 또는 KOI8-R에 대한 단말기 지원, 115



라틴-1 작성 시퀀스, 101  
라틴-2 작성 시퀀스, 102  
라틴-3, 작성 시퀀스, 105  
라틴-4 작성 시퀀스, 104  
라틴-5 작성 시퀀스, 105  
라틴-9 작성 시퀀스, 105  
라틴-n 단말기, 115

**러**  
러시아어, 문자 지원, 93

**레**  
레이아웃 서비스, 126  
레이아웃 수정자 방향, 126

**렌**  
렌더 테이블, 136

**로**  
로드, STREAMS 모듈, 114  
로캘, 37  
  개념, 25  
  기본 언어, 25  
  날짜 형식, 28  
  단어 분리자, 32  
  문자 세트, 32  
  문화적 규약, 26  
  범주, 27  
  사용 국가, 25  
  숫자 형식, 29  
  시간 형식, 28  
  정렬 순서, 32  
  코드 세트, 25  
  키보드 차이점, 35  
  통화, 23  
  통화 형식, 30  
  페이지 크기, 35  
  환경 변수, 112, 113  
로캘 종속 프로로그 파일, 151  
로캘 콘텐츠, 55, 75

로캘 환경 변수, 112

**멀**  
멀티바이트 파일 코드 변환, 43

**메**  
메시지 카탈로그, 50  
메시지 카탈로그 만들기, 50

**명**  
명령, CSI화, 38

**문**  
문자, 32  
문자 지원, 93  
문자열  
  XmTextFieldGetLayoutModifier, 130

**벨**  
벨기에어 키보드, 107

**변**  
변환, 116  
  멀티바이트 및 넓은 문자 프로세스 코드, 43

**부**  
부분 지역화 패키지, 187  
  ISO8859, 190  
  Netscape Communicator 4.7, 187  
  남미, 192  
  남부 유럽, 192  
  동유럽, 188  
  서유럽, 192

부분 지역화 패키지 (계속)  
   스웨덴어, 192  
   오스트랄라시아, 187  
   인도어, 190  
   정체 한자, 189  
   중국어, 일본어 및 한국어의 공통 파일, 187  
   중미, 188  
   타이어, 192  
   한국어, 191

## 새

새 키보드 지원, 16

## 서

서부 유럽, 문자 지원, 93  
 서부 유럽권 언어, 문자 지원, 93  
 서유럽 문자, 32

## 설

설정  
   TTY 환경, 113  
   단말기 옵션, 115

## 숫

숫자 형식  
   1000 단위 구분 기호, 29  
   구분자, 29  
   규약, 29  
   기수 문자, 29  
   소수 자릿수, 29

## 스

스웨덴어 패키지, 207  
 스칸디나비아어 및 발트어, 문자 지원, 93  
 스크립트 선택, 95  
 스페인어 패키지, 205  
 스페인어, 문자 지원, 93

## 시

시간대, 28  
 시스템 라이브러리, 응용프로그램 연결, 41

## 아

아랍어, 문자 지원, 93  
 아랍어 키보드, 106  
 아시아 언어 Solaris용 조합 로캘, 22  
 아시아권 지원 언어, 76

## 영

영어  
   문자 지원, 93  
   입력 모드, 97

## 유

유로화, 22  
 유틸리티  
   genmsg, 50

## 응

응용프로그램  
   FontSet/XmFontList 정의, 119  
   XPG4, 42  
   국제화, 119  
   시스템 라이브러리에 연결, 41  
 응용프로그램 국제화, 119  
 응용프로그램 연결, 41

## 이

이식 가능 레이아웃 서비스(PLS), 121  
   기능 목록, 121  
   대칭적 교체, 121  
   문자열 검증, 121  
   숫자 모양, 121  
   슬러, 121  
   양방향성, 121

이탈리아어 패키지, 198

**인**

인쇄 레이아웃 프롤로그 파일, 151  
인쇄 필터, 22, 143  
인쇄 필터 강화, 144  
인쇄 필터 개선 사항, 16

**일**

일광 절약 시간 (DST), 28  
일본어  
    iconv 모듈, 22, 86  
    locales, 84  
    글꼴, 84  
    문자 세트, 84  
    문자 지원, 93  
    입력 시스템, 85  
    전체 지역화 패키지, 87  
    지역화, 84  
일본어 가타카나, 32  
일본어 간지, 32  
일본어 입력 모드, 109  
일본어 텍스트  
    가타카나, 32  
    간지, 32  
    언어 소개, 32  
    히라가나, 32  
일본어 패키지, 200  
일본어 히라가나, 32

**입**

입력 모드  
    en\_US.UTF-8 로캘, 95  
    그리스어, 106, 110  
    영어, 97  
    키릴 문자, 105, 106, 107, 109, 110, 111, 112

**작**

작성 시퀀스, 106, 107, 109, 110  
    라틴-1, 98, 101

작성 시퀀스 (계속)  
    라틴-2, 102  
    라틴-3, 104, 105  
    라틴-4, 104  
    라틴-5, 105  
    라틴-9, 105

**저**

저장, STREAMS 모듈 설정, 116

**전**

전체 Solaris 로캘, 26  
전체 지역화 패키지  
    간체 한자, 193  
    공유, 211  
    독일어, 197  
    스웨덴어, 207  
    스페인어, 205  
    이탈리아어, 198  
    일본어, 87, 200  
    정체 한자, 208  
    프랑스어, 195  
    한국어, 203

**정**

정적 연결, 41  
정체 한자  
    input method, 81  
    locales, 81  
정체 한자 입력 모드, 111  
정체 한자 패키지, 208  
정체 한자(홍콩)  
    Big5+HKSCS 로캘, 22  
    UTF-8 로캘, 22

**중**

중국, 34  
중국어 hanzi, 33  
중국어 pinyin, 34  
중국어 텍스트, 언어 소개, 33



## 한

### 한국어

- iconv 모듈, 89
- input method, 87
- locales, 87
- 문자 지원, 93
- 비트맵 글꼴, 88, 89
- 지역화, 87
- 한국어 입력 모드, 109
- 한국어 텍스트
  - 언어 소개, 33
  - 한글, 33
  - 한자, 33
- 한국어 트루 타입, 89
- 한국어 패키지, 203
- 한국어의 한글, 33
- 한국어의 한자, 33

## 힌

- 힌디어, 문자 지원, 93
- 힌디어 스크립트 지원, 22
- 힌디어 텍스트, 34

## 헝

- 헝가리어, 문자 지원, 93

## 헤

- 헤브루어, 문자 지원, 93
- 헤브루어 키보드, 108

## 협

- 협정 세계시, 28

## 환

- 환경 변수, LANG, 113

## 히

- 히브리어 텍스트, 34

