



JFP ユーザーズガイド

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303
U.S.A. 650-960-1300

Part No: 805-4178-10
1998 年 11月

本製品およびそれに関連する文書は著作権法により保護されており、その使用、複製、頒布および逆コンパイルを制限するライセンスのもとにおいて頒布されます。日本サン・マイクロシステムズ株式会社による事前の許可なく、本製品および関連する文書のいかなる部分も、いかなる方法によっても複製することが禁じられます。

本製品の一部は、カリフォルニア大学からライセンスされている Berkeley BSD システムに基づいていることがあります。UNIX は、X/Open Company, Ltd. が独占的にライセンスしている米国ならびに他の国における登録商標です。フォント技術を含む第三者のソフトウェアは、著作権により保護されており、提供者からライセンスを受けているものです。

RESTRICTED RIGHTS: Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DFAR 227.7202-3(a).

本製品に含まれる HG 明朝 L と HG ゴシック B は、株式会社リコーがリョーベイマジクス株式会社からライセンス供与されたタイプフェイスマスタをもとに作成されたものです。平成明朝体 W3 は、株式会社リコーが財団法人 日本規格協会 文字フォント開発・普及センターからライセンス供与されたタイプフェイスマスタをもとに作成されたものです。また、HG 明朝 L と HG ゴシック B の補助漢字部分は、平成明朝体 W3 の補助漢字を使用しています。なお、フォントとして無断複製することは禁止されています。

Sun, Sun Microsystems, SunSoft, SunDocs, SunExpress, OpenWindows は、米国およびその他の国における米国 Sun Microsystems, Inc. (以下、米国 Sun Microsystems 社とします) の商標もしくは登録商標です。

サンロゴマークおよび Solaris は、米国 Sun Microsystems 社の登録商標です。

すべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。SPARC 商標が付いた製品は、米国 Sun Microsystems 社が開発したアーキテクチャに基づくものです。

OPENLOOK, OpenBoot, JLE は、日本サン・マイクロシステムズ株式会社の登録商標です。

Wnn は、京都大学、株式会社アステック、オムロン株式会社で共同開発されたソフトウェアです。

Wnn6 は、オムロン株式会社で開発されたソフトウェアです。(Copyright OMRON Co., Ltd. 1998 All Rights Reserved.)

ATOK は、株式会社ジャストシステムの登録商標です。

ATOK7 は株式会社ジャストシステムの著作物であり、ATOK7 にかかる著作権その他の権利は、すべて株式会社ジャストシステムに帰属します。

ATOK8 は株式会社ジャストシステムの著作物であり、ATOK8 にかかる著作権その他の権利は、すべて株式会社ジャストシステムに帰属します。

本書で参照されている製品やサービスに関しては、該当する会社または組織に直接お問い合わせください。

OPEN LOOK および Sun Graphical User Interface は、米国 Sun Microsystems 社が自社のユーザおよびライセンス実施権者向けに開発しました。米国 Sun Microsystems 社は、コンピュータ産業用のビジュアルまたはグラフィカル・ユーザインタフェースの概念の研究開発における米国 Xerox 社の先駆者としての成果を認めるものです。米国 Sun Microsystems 社は米国 Xerox 社から Xerox Graphical User Interface の非独占的ライセンスを取得しており、このライセンスは米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

DiComboBox ウィジェットと DtSpinBox ウィジェットのプログラムおよびドキュメントは、Interleaf, Inc. から提供されたものです。(Copyright (c) 1993 Interleaf, Inc.)

本書は、「現状のまま」をベースとして提供され、商品性、特定目的への適合性または第三者の権利の非侵害の黙示の保証を含みそれに限定されない、明示的であるか黙示的であるかを問わない、なんらの保証も行われぬものとします。

本製品が、外国為替および外国貿易管理法 (外為法) に定められる戦略物資等 (貨物または役務) に該当する場合、本製品を輸出または日本国外へ持ち出す際には、日本サン・マイクロシステムズ株式会社の事前の書面による承諾を得ることのほか、外為法および関連法規に基づく輸出手続き、また場合によっては、米国商務省または米国所轄官庁の許可を得ることが必要です。

© 1998 by Sun Microsystems, Inc.



目次

はじめに	xi
パートI JFP - SunOS	
1. JFP の概要	3
JFP とは	3
各国語対応と日本語対応	3
国際化	4
CSI (Code Set Independence)	4
JFP の位置づけ	4
JFP がインストールされていることの確認方法	4
日本語ロケール	5
日本語化された SunOS メッセージ	5
日本語環境用 SunOS のコマンド	5
iconv コード変換機能用の日本語モジュール	6
STREAMS モジュールによる端末用コード変換	7
日本語プリンタのサポート	7
日本語入力機能	7
日本語キーボードのサポート	8
日本語オンラインマニュアル	8
JLE バイナリ互換性サポートパッケージ	8

- 2. 日本語ロケール 9
 - 日本語文字コード 9
 - 日本語 EUC 9
 - PC 漢字コード 10
 - UTF-8 11
 - 日本語ロケール 12
 - 文字分類 (LC_CTYPE カテゴリ) 12
 - 文字の照合順序 (LC_COLLATE カテゴリ) 21
 - 単語処理 22
 - 日付と時刻の表示形式 (LC_TIME カテゴリ) 22
- 3. 日本語対応 **SunOS** のユーザー環境設定 25
 - 環境変数によるロケールの設定 25
 - 端末の設定 26
 - コマンド検索パスの設定 28
 - オンラインマニュアル検索パスの設定 30
 - 初期化ファイル利用のヒント 31
 - .login 31
 - .cshrc 32
 - .profile 32
 - Kornシェル起動時コマンド実行ファイル 33
- 4. 日本語対応 **SunOS** のシステム環境設定 35
 - システムロケールの設定 35
 - 電子メールの送受信に関する設定 36
 - in.comsat の設定 36
 - 日本語プリンタに関する設定 36
 - 日本語 dumb プリンタのサポート 36
 - 日本語 PostScript プリンタのサポート 39
 - プリンタの追加方法 41

	JIS X 0212、ユーザー定義文字、ベンダー定義文字の印刷サポート	51
	日本語キーボードのサポート	53
	日本語タイプ 4、5 キーボード	54
5.	国際化 SunOS コマンドの利用	55
	vi エディタ	55
	vi の起動	56
	テキストの挿入 (文字幅の処理、行末処理)	57
	1 文字の置換、削除、大文字・小文字変換	59
	単語単位の日本語処理	60
	改行マージンと連結コマンド (行の分割と連結)	63
	日本語文字を含む文字列の検索と置換	65
	ファイルの保存と vi の終了	67
	オンラインマニュアルの参照	69
	マニュアルページの参照	69
	清書済みオンラインマニュアルと windex の作成	70
	whatis による機能要約の参照	71
	apropos によるキーワード検索	71
	電子メールの送受信	71
	メールの起動	72
	メッセージの作成	72
	メール作成の中止	75
	送信したメッセージの保存	75
	メッセージの読み取り	76
	到着したメッセージの保存	77
	到着したメッセージの編集	78
	メッセージの返答方法	79
	メール到着の通知	81
6.	日本語環境用 SunOS コマンド	83

フィルタを用いたコード変換	83
漢字端末と日本語入力	87
日本語文字コード変換 STREAMS モジュール	88
mailx/in.comsat でのコード変換	90
パートII JFP - GUI	
7. GUI 環境の起動	93
ログインサーバー	93
セッション	94
言語	94
8. Solaris CDE と日本語 OpenWindows 環境に共通の機能	97
日本語入力システムの切り替え方法	97
日本語入力サーバー(htt)	98
日本語フォント	99
日本語 TrueType フォント	99
日本語ビットマップフォント	100
JIS X 0208-1990 文字集合	100
JIS X 0212-1990 補助漢字	101
IBM 拡張文字と NEC 特殊文字	101
ユーザー定義文字フォント	101
XGL 漢字ストロークフォント	101
フォントのディレクトリ構造	101
EUC コードセット	102
ja_JP.UTF-8 ロケール	102
ja と ja_JP.PCK ロケール間の文字変換	102
日本語化されているプログラム	103
インストールツール	103
システム管理ツール	103
グラフィックスライブラリ	103

	Kodak Color Management System (KCMS)	103
	AccessX	103
	Motif	103
	電源管理	104
	xserverconfig	104
	ToolTalk	104
	仮想 LED	104
9.	Solaris CDE の機能概要	105
	日本語化されているアプリケーション	105
	日本語環境固有のアプリケーション	106
	日本語オンラインヘルプ	107
	日本語 AnswerBook	107
	日本語入力システム	108
	デスクトップ環境のカスタマイズ	108
	フォントグループの設定方法	108
	プリンタの設定方法	109
	Solaris CDE を使用する際の推奨環境	110
10.	日本語 OpenWindows の機能概要	111
	日本語化されているプログラム	111
	日本語 DeskSet プログラム	111
	OPENLOOK ウィンドウマネージャ	112
	ワークスペースプロパティとワークステーション情報	112
	ツールキットのライブラリ	112
	ボリューム管理ユーザーインタフェース	113
	日本語環境固有のアプリケーション	113
	cs00 ユーザー辞書ツール	113
	ワークスペースプロパティを使った環境の切り替え	114
	日本語オンラインヘルプ	114

- 日本語 Magnify Help 114
- 日本語ヘルプハンドブック 114
- 日本語オンラインマニュアル 115
- 日本語フォントの使い方 115
 - XView アプリケーション 115
 - OLIT アプリケーション 116
- 日本語対応ツールキットによる日本語プログラムの開発環境 117
 - XView 117
 - OLIT 117
- 11. 日本語 **OpenWindows** の環境設定 119
 - 初期設定と起動 119
 - 日本語 OpenWindows 2.0.1 環境からの移行 120
 - キーの機能定義 121
 - 日本語 OpenWindows 3.2 以前の環境からの移行 121
- 12. 日本語 **OpenWindows** のカスタマイズ 123
 - ワークスペース 123
 - 基本ロケール 125
 - その他の設定 125
 - ロケールの再定義 126
 - EUC コードセット 3 (JIS X 0212) 126
- 13. 仮想 **LED** 127
 - 起動方法 127
 - オプション 128
 - unconditional 128
 - kanaonly 128
 - Wi (iconic) 128
 - display host:x.x 129
 - Ws *width height* 129

-Wp *x y* 129
-WP *x y* 129
-geometry (WxH){+|-}X{+|-}Y 129
-icon_geometry (WxH){+|-}X{+|-}Y 129
-fg (-foreground) #rrggbg | *color_name* 129
-bg (-background) #rrggbg | *color_name* 130
-brightcolor #rrggbg | *color_name* 130
-darkcolor #rrggbg | *color_name* 130
-intervaltime *usec* 130

14. ユーザー定義文字の登録 (**Solaris** 外字ツール) 131

機能概要 132

文字エディタ 133

一覧表 134

参照表 135

起動方法 135

各機能の説明 136

文字エディタ 136

一覧表 147

参照表 149

その他の機能 151

ユーザー定義文字の日本語入力システムへの登録方法 153

ユーザー定義文字のアプリケーションでの利用方法 156

日本語入力システム Wnn6 を利用している場合 157

日本語入力システム ATOK8 を利用している場合 158

日本語入力システム cs00 を利用している場合 158

ユーザー定義文字の DPS 上での利用方法 159

ユーザー定義文字の印刷方法 160

オプション 160

- 環境設定 161
 - 環境変数 161
 - リソース 161
- 15. フォントの移行 163**
 - Solaris1.x SunView で使用していた evfont の再利用 163
 - evfont から Adobe/MIT BDF 2.1 への変換 163
 - Adobe/MIT BDF 2.1 から PCF フォントファイルへの変換 164
 - フォントエイリアスの追加 164
 - フォントセットの定義 165
 - Solaris 2.5.1 以前のシステムで作成したユーザー定義文字の移行 165
 - フォントエディタで作成したビットマップフォントを移行する場合 166
 - フォントマネージャで作成したフォントを移行する場合 167
 - フォントエディタで作成したビットマップフォントとフォントマネージャで作成したフォントをマージする場合 168
- A. 文字コード変換規則 171**
 - 新しい文字コード変換規則 171
 - IBM 拡張文字の変換表 173
 - 索引 179

はじめに

このマニュアルでは、Solaris™ における日本語処理機能 JFP (Japanese Feature Package) について説明します。Solaris は、日本語対応の SunOS™ と Solaris 共通デスクトップ環境 (Common Desktop Environment) (以降 Solaris CDE とします) または日本語 OpenWindows™ から構成されています。Part 1 では日本語対応の SunOS に関する内容を中心に JFP 全般について説明します。また、Part 2 では日本語 GUI について説明します。個々の項目は日本語処理機能についてだけ書かれています。各国共通の機能については、『OpenWindows ユーザーズガイド』、『OpenWindows ユーザーズガイド (上級編)』、および『プログラミングの国際化』を参照してください。また、国際化のプログラミングや日本語化のプログラミングに関しては、『プログラミングの国際化』と『JFP 開発ガイド』を参照してください。

対象読者

このマニュアルは、Solaris に関する基礎知識を持っている方を対象としています。また、このマニュアルを使用するには、Sun™ システムが Solaris 2.6 オペレーティング環境およびその互換バージョンで動作している必要があります。

内容の紹介

第 1 章

JFP の位置づけを紹介し、JFP の提供する機能について SunOS を中心に説明します。

第 2 章

JFP で提供している日本語ロケールの定義内容に関して説明します。

第 3 章

日本語をサポートできる SunOS に対して行う個々のユーザーの環境設定について説明します。

第 4 章

システムで定義済みの SunOS の日本語環境を、ユーザーが必要に応じて変更する方法と注意事項について説明します。

第 5 章

国際化された SunOS コマンドを日本語環境で利用する方法を vi や mailx を中心に説明します。

第 6 章

JFP によって日本語環境用に提供される SunOS コマンドについて説明します。

第 7 章

GUI 環境の起動に関して説明します。

第 8 章

Solaris CDE と日本語 OpenWindows 環境に共通する機能について説明します。

第 9 章

Solaris CDE の主な機能について説明します。

第 10 章

日本語 OpenWindows 3.6 の主な機能について説明します。

第 11 章

日本語 OpenWindows の初期設定、起動方法などについて説明します。

第 12 章

日本語 OpenWindows でのカスタマイズについて説明します。

第 13 章

仮想 LED の機能と使い方について説明します。

第 14 章

ユーザー定義文字の登録について説明します。

第 15 章

既に利用しているフォントの移行方法について説明します。

付録 A

文字コード変換規則について説明し、さらに IBM 拡張文字の変換表も記載します。

読んでおく必要があるマニュアル

- 『Solaris 7 ご使用にあたって (SPARC 版)』または『Solaris 7 ご使用にあたって (Intel 版)』
- 『Solaris 7 オンラインリリース情報』
- 『OpenWindows ユーザーズガイド』
- 『OpenWindows ユーザーズガイド (上級編)』

関連マニュアル

- 『JFP 開発ガイド』
- 『プログラミングの国際化』
- 『SunOS リファレンスマニュアル』

マニュアルの注文方法

SunDocs™ プログラムでは、米国 Sun Microsystems™, Inc. (以降、Sun™ とします) の 250 冊以上のマニュアルを扱っています。このプログラムを利用して、マニュアルのセットまたは個々のマニュアルをご注文いただけます。

マニュアルのリストと注文方法については、米国 SunExpress™, Inc. のインターネットホームページ <http://www.sun.com/sunexpress> にあるカタログセクションを参照してください。

表記上の規則

このマニュアルでは、次のような字体や記号を特別な意味を持つものとして使用します。

表 P-1 表記上の規則

字体または記号	意味	例
AaBbCc123	コマンド名、マニュアルページ名、ファイル名、およびディレクトリ名を示します。または、画面上のコンピュータ出力を示します。	.login ファイルを編集します。ls -a を使用してすべてのファイルを表示します。 sun%
AaBbCc123	ユーザーが入力する文字を、画面上のコンピュータ出力とは区別して示します。	sun% su Password:
<i>AaBbCc123</i> または <あいうえお>	コマンド行のプレースホルダを示します。プレースホルダは、実際に使用する場合には特定の名前または値で置き換えます。	ファイルを削除するには、rm <i>filename</i> と入力します。 ファイルを削除するには、rm <ファイル名> と入力します。
『	参照する書名を示します。	『SunOS リファレンスマニュアル』
「	参照する章、節を示します。または、語を強調する場合、画面上の項目を表す場合に使用します。	第 5 章「衝突の回避」を参照してください。 「特定」の機器でのみ使用します。

ただし AnswerBook2™ では、ユーザーが入力する文字と画面上のコンピュータ出力は区別して表示されません。

マウスボタンの使い方、ウィンドウメニューの使い方については、*OpenWindows* ユーザーズガイドの「はじめに」を参照してください。

コード例はボックスで囲んで示します。例えば、次のように表示されます。

■ C シェルプロンプト

```
system%
```

■ Bourne シェルプロンプト

```
system$
```

■ スーパーユーザーのプロンプト

```
system#
```

- キーボードのキー名は英文で、頭文字を大文字で示します (例 : Shift キーを押す)。ただし、キーボードによっては Enter キーが Return キーの動作をします。
- ダッシュ (-) は 2 つのキーを同時に押すことを示します。たとえば、Ctrl-D は Control キーを押したまま D キーを押すことを意味します。

一般規則

- 「x86」という用語は、一般に Intel 8086 ファミリに属するマイクロプロセッサを意味します。これには、Pentium、Pentium Pro の各プロセッサ、および AMD と Cyrix が提供する互換マイクロプロセッサチップが含まれます。このマニュアルでは、このプラットフォームのアーキテクチャ全体を指すときに「x86」という用語を使用し、製品名では「Intel 版」という表記で統一しています。

画面例について

このマニュアルで使用する画面例と、Solaris 7 の画面例とは、ヘッダ表示などに若干の違いがある場合があります。

パート I **JFP - SunOS**



JFP の概要

この章では Japanese Feature Package (以降、JFP とします) の位置づけなどを紹介し、JFP の提供する機能について SunOS を中心に説明します。Solaris CDE および日本語 OpenWindows の機能については、Part 2 - 「JFP - GUI」を参照してください。

JFP とは

JFP は、Solaris で日本語機能を使用するための追加パッケージであり、SunOS と Solaris CDE、OpenWindows 用の日本語サポート機能を含みます。JFP は国際化された SunOS、Solaris CDE、OpenWindows とともに Solaris を構成します。

各国語対応と日本語対応

システムのソフトウェアおよびアプリケーションを特定の言語や文化、慣習 (これらをまとめてロケール (locale) と呼びます) に適応させることを、このマニュアルでは各国語対応 (localization) と呼ぶことにします。日本語および日本の文化、慣習を対象とした各国語対応が日本語対応です。各国語対応の要求はさまざまですが、そのある部分は言語に依存しない共通の枠組みで処理でき、またある部分は特定の言語専用の機能を必要とします。

国際化

各国語対応のために利用できる共通の枠組みを提供することを国際化 (internationalization) と呼びます。システムおよびアプリケーションを国際化することにより、複数の言語に対する各国語対応が、より簡単かつ統一された方法で実現できます。

CSI (Code Set Independence)

エンコーディングに依存しない国際化機構を、CSI (Code Set Independence) 化された国際機構と呼びます。国際化機構が CSI 化されたことにより、日本語 EUC (Extended Unix Code) を扱う ja ロケールに加え、PC 漢字コードを扱う ja_JP.PCK ロケール、UTF-8 コードを扱う ja_JP.UTF-8 ロケールがサポートされます。ただし、現時点では、対応するコマンドやライブラリなどに制限があります。CSI 化されたコマンドやライブラリなどに関しては、『プログラミングの国際化』を参照してください。

JFP の位置づけ

JFP は国際化された SunOS と Solaris CDE および OpenWindows に対して日本語のための各国語化機能を提供します。

JFP がインストールされていることの確認方法

JFP は、SUNWjfp、SUNWjfpu などのパッケージと呼ばれるインストールの単位の集合からなります。JFP のパッケージがシステム上に存在するかどうかは、pkginfo(1) コマンドを使って調べることができます。たとえば、grep(1) コマンドを併用して次のように入力してください。

```
sun% pkginfo | grep SUNWjfp
:
system    SUNWjfp   Japanese Feature Package root files
system    SUNWjfpu  Japanese Feature Package user files
:
:
```

「ご使用にあたって」の付録に JFP パッケージの一覧が掲載されていますので、特定の使いたい機能がインストールされているかどうかこの付録を参照してパッケージ名を調べてから `pkginfo` で確認してください。

日本語ロケール

JFP では、システム提供の日本語ロケールとして、日本語 EUC を扱う `ja` ロケール、PC 漢字コードを扱う `ja_JP.PCK` ロケールおよび UTF-8 を扱う `ja_JP.UTF-8` ロケールを提供します。また、日本語 Solaris 1.1.x システムとのネットワーク環境での共存のため「`japanese`」というロケール名も `ja` と同等のロケールを提供します。日本語ロケールの定義内容の詳細は、第 2 章を参照してください。

日本語化された SunOS メッセージ

JFP は、メッセージの国際化機能を利用して、日本語化されたメッセージカタログを提供しています。ただし、SunOS が提供しているすべてのコマンド、ライブラリが、メッセージの国際化を行なっているわけではありません。それらのメッセージは日本語環境においても英語で表示されます。

日本語環境用 SunOS のコマンド

SunOS 5.x で国際化されたコマンドに加えて、JFP では日本語環境をサポートするために以下の SunOS コマンドを提供します。これらのコマンドの使用方法に関しては、第 6 章および各コマンドのマニュアルページを参照してください。

表 1-1 JFP 日本語環境用コマンド

```

/usr/bin/
euctoibmj 日本語 EUC から IBM 本語コードへのコード変換フィルタ
evftobdf evfont と BDF の間のフォントフォーマット変換コマンド
euctojis 日本語 EUC から JIS コードへのコード変換フィルタ
euctosj 日本語 EUC から PC 漢字 (シフト JIS) コードへのコード変換フィルタ
evftobdf SunView™ で使用していた evfont を BDF フォントに変換 (SPARC™ のみ)
ibmjtoeuc IBM 漢字コードから日本語 EUC へのコード変換フィルタ
jistoec JIS コードから日本語 EUC へのコード変換フィルタ
jistosj JIS コードから PC 漢字コードへのコード変換フィルタ
jtops PostScript™ プリンタに日本語テキストを出力するためのフィルタ
jtty 日本語端末用 STREAMS モジュールの制御コマンド
kanji JIS X0208、JIS X0212、およびベンダー定義文字の文字コード一覧表示
sjtoec PC 漢字コードから日本語 EUC へのコード変換フィルタ
sjtojis PC 漢字コードから JIS コードへのコード変換フィルタ

```

また、JFP は、SunOS の国際化されたコマンドを拡張した、以下のコマンドを提供します。これらのコマンドの使用法に関しては、第 5 章、第 6 章、および各コマンドのマニュアルページを参照してください。

表 1-2 国際化拡張コマンド

```

/usr/bin/
  setterm コードセット変換用 STREAMS モジュールの構成用コマンド
/usr/SUNWale/bin/
  mailx 複数バイト言語用国際化のために拡張された mailx
  talk 複数バイト言語用国際化のために拡張された talk
/usr/SUNWale/sbin/
  in.comsat 複数バイト言語用国際化のために拡張された in.comsat

```

注・mailx、talk、および install_comsat は、ja ロケールでのみ動作が保証されています。

iconv コード変換機能用の日本語モジュール

JFP では、国際化 SunOS で提供されているコード変換機能 iconv の日本語コードセットのサポートとして、この機能で使用する日本語モジュールを提供します。5 ページの「日本語環境用 SunOS のコマンド」で紹介している、euctosj、sjtojis などのコード変換はベンダー定義文字やユーザー定義文字の変換をサポートしてい

ないため、これらの定義文字を含めた変換を考慮する場合は、`iconv` を使用してください。コード変換機能 `iconv(1)` の用法に関しては、第 6 章を、`iconv(3)` の用法に関しては、『JFP 開発ガイド』を参照してください。

STREAMS モジュールによる端末用コード変換

JFP は、各種日本語端末からの `tty` ベースの入出力に対して、コードセット変換を行う STREAMS モジュール群を提供します。

日本語端末から STREAMS モジュールを使用するときには、国際化された SunOS コマンドとして提供されている `setterm(1)` を使用します。STREAMS モジュールの用法に関しては、第 6 章および各コマンドのマニュアルページを参照してください。

日本語プリンタのサポート

JFP では、LP 印刷サービスを使用してドットマトリックス漢字プリンタ (EPSON VP-5085、NEC PC-PR201)、日本語ページプリンタ (Canon LASERSHOT)、日本語 PostScript プリンタで日本語テキストを印刷するために、これらの変換フィルタ群を提供します。日本語プリンタの用法に関しては、第 4 章を参照してください。

日本語入力機能

JFP では、`Wnn6`、`ATOK8`、および `cs00` により実現される日本語入力機能を提供しています。詳細は、『日本語入力システムの概要とセットアップ』、『`Wnn6` ユーザーズガイド』、『`ATOK8` ユーザーズガイド』、および『`cs00` ユーザーズガイド』を参照してください。

日本語キーボードのサポート

JFP では、次の日本語キーボードをサポートしています。

表 1-3 日本語キーボードのサポート

プロセッサの種類	キーボードの種類
SPARC	日本語タイプ 4 キーボード
	日本語タイプ 5 キーボード
x86	106 キーボード
	J3100 キーボード

日本語オンラインマニュアル

JFP では、日本語のオンラインマニュアルを提供しています。ただし、すべてのオンラインマニュアルが日本語化されているわけではありません。

JLE バイナリ互換性サポートパッケージ

JEP では、JLE システム上作成されたアプリケーションを動作させるために、JLE バイナリ互換性サポートパッケージを提供しています。バイナリ互換性パッケージの一般的な注意事項および使用条件に関しては、「バイナリ互換性ガイド」を参照して下さい。以下にパッケージ固有の注意事項を示します。

- JLE ロケールデータベースは JFP のものとは互換性がないため、JLE バイナリ互換性サポートパッケージでは `/usr/lib/oldlocale` に JLE 互換ロケールデータベースを置いています。JLE アプリケーションは自動的にこのディレクトリを参照するようになっています。

日本語ロケール

この章では、JFP で提供している日本語ロケールの定義内容に関して説明します。

日本語文字コード

日本語 EUC

日本語 EUC は、EUC (Extended UNIX Code : 拡張 UNIX コード) に、以下の日本語文字集合を割り当てた文字集合およびエンコーディングを提供するものです。ja ロケールで日本語を表現する文字コード体系として使われています。

表 2-1 日本語 EUC の文字集合

文字集合 (符号化文字集合)	文字コード
JIS X 0201-1976	
機能キャラクタ集合	0x00 - 0x19
間隔文字	0x20
ローマ文字用図形キャラクタ集合	0x21 - 0x7e
抹消文字	0x7f
ISO 6429 C1 制御文字 (0x8e、0x8f を除く)	0x80 - 0x9f
JIS X 0201-1976 片仮名用図形キャラクタ集合 (文字未定義領域 E/0 - F/14 は含まない)	0x8ea1 - 0x8edf

表 2-1 日本語 EUC の文字集合 続く

文字集合 (符号化文字集合)	文字コード
JIS X 0208-1990 (1 - 84 区)	0xa1a1 - 0xf4fe
JIS X 0212-1990 (1 - 84 区)	0x8fa1a1 - 0x8ff4fe
ユーザー定義文字 (1 - 20 区)	0xf5a1 - 0xfefe 0x8ff5a1 - 0x8ffefe

ただし、JIS で未定義となっている以下の領域には、ベンダー定義文字を割り当てています。

JIS X 0208 13 区	特殊記号
JIS X 0212 83 区 - 84 区	JIS X 0212 に含まれない IBM 拡張文字

日本語 EUC の場合、ベンダー定義文字とは、JIS X 0208 13 区文字未定義領域に割り当てられた特殊記号、ならびに JIS X 0212 83 区 - 84 区文字未定義領域に割り当てられた JIS X 0208 と JIS X 0212 のどちらにも含まれない IBM 拡張文字を表します。

PC 漢字コード

PC 漢字コード (以降、PCK とします) は一般に「シフト JIS (あるいは MS 漢字) コード」と呼ばれ、Microsoft が Windows 3.1 で規定したマイクロソフト標準キャラクタセットと同等の文字集合およびエンコーディングを提供するものです。ja_JP.PCK ロケールで日本語を表現する文字コード体系として使われています。

表 2-2 PCK の文字集合

文字集合 (符号化文字集合)	文字コード
JIS X 0201-1976	
機能キャラクタ集合	0x00 - 0x19
間隔文字	0x20
ローマ文字用図形キャラクタ集合	0x21 - 0x7e
抹消文字	0x7f

表 2-2 PCK の文字集合 続く

文字集合 (符号化文字集合)	文字コード
JIS X 0201-1976 片仮名用図形キャラクタ集合 (文字未定義領域 E/0 - F/14 は含まない)	0xa1 - 0xdf
JIS X 0208-1990 (1 - 84 区)	0x8140 - 0x9ffc 0xe040 - 0xeafc
NEC 選定 IBM 拡張文字	0xed40 - 0xeffc
ユーザー定義文字 (1 - 20 区)	0xf040 - 0xf9fc
IBM 拡張文字	0xfa40 - 0xfcfc

ただし、JIS で未定義となっている以下の領域には、ベンダー定義文字を割り当てています。

JIS X 0208 13 区	特殊記号
-----------------	------

PCK の場合、ベンダー定義文字とは、JIS X 0208 13 区文字未定義領域に割り当てられた特殊記号、NEC 選定 IBM 拡張文字、ならびに IBM 拡張文字を表します。

UTF-8

UTF-8 は、Unicode のエンコーディングの 1 つであり、ASCII 部分 (0x00-0x7F) に関しては EUC または PCK と互換性をたもったまま、UCS-2 や UCS-4 の文字を使えるようにしたエンコーディングです。ja_JP.UTF-8 ロケールで文字を表現する文字コード体系として使われています。

表 2-3 UTF-8 の文字集合

文字集合 (符号化文字集合)	文字コード
Unicode 2.1	0x00 - 0xefbfbf (U+0000 - U+FFFF)

ユーザー定義文字領域は Unicode 2.1 で Private Use area に割り当てられている 0xee8080 - 0xefa3bf (U+E000 - U+F8FF) が該当します。日本語 EUC や PCK で表現できるすべての文字（ベンダー定義文字も含む）も Unicode 2.1 の中に含まれています。

注 - U+XXXX の XXXX は Unicode 2.1 で表現しているコードを意味します。

注 - Unicode 2.1 で定義された Surrogate Area (U+D800 - U+DFFF) はサポートされてません。

注 - Solaris のフォントの制限で、日本語 UTF-8 で Unicode 2.1 で定義されたすべての文字が表示できるわけではありません。

日本語ロケール

JFP では、システム提供の日本語ロケールとして、日本語 EUC を扱う ja ロケール、PCK を扱う ja_JP.PCK ロケールおよび UTF-8 を扱う Ja_JP.UTF-8 ロケールを提供します。また、JLE 1.x システムとのネットワーク環境での共存のため「japanese」というロケール名も ja と同等のロケールとして提供します。

ロケールとは、特定の言語や文化、慣習のことを意味します。日本語ロケールデータベースを作成することによって、国際化されたシステムソフトウェアやアプリケーションを、ロケールを切り替えるだけで日本語環境に適応できます。

基本的に日本語ロケール間の違いは、扱う文字コードが異なるだけで、ロケールとしての挙動は同じになるように定義されています。たとえば、ja ロケールで print という文字分類クラスに属する文字は、ja_JP.PCK や ja_JP.UTF-8 ロケールでも同じ文字が定義されています。ただし、日本語 EUC の方が PCK より、UTF-8 の方が日本語 EUC より扱う文字が多いので、UTF-8 にしか存在しない文字に関しては、ja_JP.UTF-8 ロケールのみ、その文字の挙動が定義されています。

文字分類 (LC_CTYPE カテゴリ)

特定の似かよった文字の集合 (クラスと呼びます) を作ることによって、プログラム上、そのクラスを意識するだけで (そのクラスにどの文字が含まれているかを意識す

ることなく)、ロケールに依存しない国際化プログラミングが可能になります。日本語ロケールでは、以下のようなクラスが定義されています。

これらの文字分類クラスは、`regex(5)` をサポートしたコマンドで、文字クラス表現の文字列として使用可能です。また、`wctype(3C)` の文字クラスとしても使用可能です。なお、文字分類は `LC_CTYPE` カテゴリの値に影響を受けます。開発者として文字分類クラスを利用する場合は、『JFP 開発ガイド』を参照してください。

表 2-4 X/Open で規定された文字分類クラス名と日本語ロケールでの定義範囲

文字クラス名	意味 / 範囲 ¹
upper	任意の大文字を表す文字クラス
	<hr/> JIS X 0201 ローマ文字用図形キャラクタ集合 アルファベット大文字 (C/1 - D/10) JIS X 0208 ローマ字大文字 (3/33 - 3/58) ギリシャ文字大文字 (6/1 - 24) ロシア文字大文字 (7/1 - 33) JIS X 0212 ² ダイアクリティカルマーク付きギリシャアルファベット文字大文字 (6/65 - 69、71、73、74、76) キリル系アルファベット大文字 (7/34 - 46) ラテン系アルファベット大文字 (9/1、2、4、6、8、9、11、12、13、15、16) ダイアクリティカルマーク付ラテン系アルファベット大文字 (10/01 - 24、26 - 87)
lower	任意の小文字を表す文字クラス
	<hr/> JIS X 0201 ローマ文字用図形キャラクタ集合 アルファベット小文字 (E/1 - F/10) JIS X 0208

表 2-4 X/Open で規定された文字分類クラス名と日本語ロケールでの定義範囲 続く

文字クラス名	意味 / 範囲 ¹
	ローマ字小文字 (3/65 - 90) ギリシャ文字小文字 (6/33 - 56) ロシア文字小文字 (7/49 - 81) JIS X 0212 ² ダイアクリティカルマーク付きギリシャアルファベット文字小文字 (6/81 - 92) キリル系アルファベット小文字 (7/82 - 94) ラテン系アルファベット小文字 (9/33 - 48) ダイアクリティカルマーク付ラテン系アルファベット小文字 (11/1 - 27、29 - 35、37 - 87)
digit	10 進表現に用いる 0 - 9 までの数字を判別するクラス JIS X 0201 ローマ文字用図形キャラクタ集合 数字 (B/0 - 9)
space	空白を判別するクラス JIS X 0201 制御キャラクタ集合 空白 (A/9 - 13) 間隔文字 JIS X 0208 間隔 (1/1)
punct	記号、特殊文字などを判別するクラス JIS X 0201 ローマ文字用図形キャラクタ集合 A/1 - 15、B/10 - C/0、D/11 - E/0、F/11 - 14
cntrl	制御文字を判別するクラス

表 2-4 X/Open で規定された文字分類クラス名と日本語ロケールでの定義範囲 続く

文字クラス名	意味 / 範囲 ¹
	<p>JIS X 0201 制御キャラクタ集合</p> <p>すべての文字</p> <p>抹消文字</p> <p>C1 制御文字</p> <p>すべての文字</p>
blank	<p>フィールド区切り文字を判別するクラス</p> <hr/> <p>JIS X 0201 制御キャラクタ集合</p> <p>A/9</p> <p>間隔文字</p> <p>JIS X 0208</p> <p>間隔 (1/1)</p>
xdigit	<p>16 進表現に用いる英数字を判別するクラス</p> <hr/> <p>JIS X 0201 ローマ文字用図形キャラクタ集合</p> <p>数字 (B/0 - 9)</p> <p>A - F、a - f (C/1 - 6、E/1 - 6)</p>
alpha	<p>アルファベットを判別するクラス</p> <hr/> <p>upper クラスと lower クラスの文字</p>
print	<p>表示可能文字を判別するクラス</p> <hr/> <p>JIS X 0201 ローマ文字用図形キャラクタ集合</p> <p>間隔文字</p> <p>JIS X 0201 片仮名用図形キャラクタ集合</p> <p>文字未定義領域以外のすべての文字</p>

表 2-4 X/Open で規定された文字分類クラス名と日本語ロケールでの定義範囲 続く

文字クラス名	意味 / 範囲 ¹
	JIS X 0208 文字未定義領域以外のすべての文字 JIS X 0212 ² 文字未定義領域以外のすべての文字 ベンダー定義文字領域 vdc クラスのうち、文字未定義領域以外のすべての文字 ユーザー定義文字領域 udc クラスのうち、文字未定義領域も含むすべての文字
graph	図形文字を判別するためのクラス print クラスから space クラスに含まれる文字を除いたすべての文字

1. JIS X 0201 機能キャラクタ集合、ローマ文字用図形キャラクタ集合、片仮名用図形キャラクタ集合での XX/YY は XX 列 YY 行を意味します。JIS X 0208、JIS X 0212 での XX/YY は XX 区 YY 点を意味します。
2. JIS X 0212 の文字に関しては、ja および ja_JP.UTF-8 ロケールのみ該当します。

注 - ja_JP.UTF-8 ロケールではそれぞれのクラスで、これら以外のに Unicode で定義された文字が追加されています。

表 2-5 日本語ロケール専用の文字分類クラス名とその定義範囲

文字クラス名	意味 / 範囲 ¹
jkanji	漢字 (漢字表記のために使用する記号、表意文字) を判別するクラス JIS X 0208 16 区 - 84 区の文字定義領域 JIS X 0212 ²

表 2-5 日本語ローケル専用の文字分類クラス名とその定義範囲 続く

文字クラス名	意味 / 範囲 ¹
	16 区 - 77 区の文字定義領域
jkata	片仮名を判別するクラス JIS X 0208 5/1 - 86、1/11、12、19、20
jhira	平仮名を判別するクラス JIS X 0208 4/1 - 83、1/11、12、21、22、26
jdigit	digit に含まれる以外の数字を判別するクラス JIS X 0208 3/16 - 25
jparen	括弧などに用いるための文字を判別するクラス JIS X 0208 1/38 - 59
line	罫線素片を判別するためのクラス JIS X 0208 8/1 - 32
jisx0201r	JIS X 0201 片仮名用図形キャラクタ集合に属する文字を判別するクラス JIS X 0201 片仮名用図形キャラクタ集合 A/1 - D/15 までのすべての文字

表 2-5 日本語ロケール専用の文字分類クラス名とその定義範囲 続く

文字クラス名	意味 / 範囲 ¹
jisx0208	JIS X 0208 に属する文字を判別するクラス <hr/> JIS X 0208 文字未定義領域の文字を含むすべての文字。ただし、1 - 84 区まで (13 区 ベンダー定義文字領域もここに含まれる)
jisx0212	JIS X 0212 ² に属する文字を判別するクラス <hr/> JIS X 0212 文字未定義領域の文字を含むすべての文字。ただし、1 - 84 区まで (83、84 区ベンダー定義文字領域もここに含まれる) ja_JP.PCK ロケールではどの文字もこのクラスに属さない
udc	ユーザー定義文字を判別するクラス <hr/> ユーザー定義文字領域の文字未定義領域も含むすべての文字 ja ロケール ユーザー定義文字 (1 - 20 区) 0xf5a1 - 0xfefe 0x8ff5a1 - 0x8ffefe ja_JP.PCK ロケール ユーザー定義文字 (1 - 20 区) ja_JP.UTF-8 ロケール ユーザー定義文字 (6400 文字) 0xee8080 - 0xefa3bf (U+E000 - U+F8FF)

表 2-5 日本語ロケール専用の文字分類クラス名とその定義範囲 続く

文字クラス名	意味 / 範囲 ¹										
vdc	<p>ベンダー定義文字を判別するクラス</p> <hr/> <p>ベンダー定義文字領域の文字未定義領域も含むすべての文字</p> <p>ja ロケール</p> <table data-bbox="730 630 1347 735"> <tr> <td>JIS X 0208 13 区</td> <td>特殊記号</td> </tr> <tr> <td>JIS X 0212² 83 区 - 84 区</td> <td>JIS X 0212 に含まれない IBM 拡張文字</td> </tr> </table> <p>ja_JP.PCK ロケール</p> <table data-bbox="730 819 1347 945"> <tr> <td>JIS X 0208 13 区</td> <td>特殊記号</td> </tr> <tr> <td>NEC 選定 IBM 拡張文字</td> <td>0xed40 - 0xeffc</td> </tr> <tr> <td>IBM 拡張文字</td> <td>0xfa40 - 0xfcfc</td> </tr> </table> <p>ja_JP.UTF-8ロケール</p> <p>定義なし</p>	JIS X 0208 13 区	特殊記号	JIS X 0212 ² 83 区 - 84 区	JIS X 0212 に含まれない IBM 拡張文字	JIS X 0208 13 区	特殊記号	NEC 選定 IBM 拡張文字	0xed40 - 0xeffc	IBM 拡張文字	0xfa40 - 0xfcfc
JIS X 0208 13 区	特殊記号										
JIS X 0212 ² 83 区 - 84 区	JIS X 0212 に含まれない IBM 拡張文字										
JIS X 0208 13 区	特殊記号										
NEC 選定 IBM 拡張文字	0xed40 - 0xeffc										
IBM 拡張文字	0xfa40 - 0xfcfc										
jalpha	<p>漢字を表す文字を判別するクラス</p> <hr/> <p>JIS X 0208</p> <p>3/33_58,、 3/65_90</p>										

表 2-5 日本語ロケール専用の文字分類クラス名とその定義範囲 続く

文字クラス名	意味 / 範囲 ¹
jspecial	特殊文字を表す文字を判別するクラス <hr/> JIS X 0208 1/2 - 94、2/1 - 14、2/26 - 33、2/42 - 48、2/60 - 74、2/82 - 89、94 JIS X 0212 2/15 - 25、2/34 - 36、2/75 - 81 JIS X 0208 13 区.....特殊記号 IBM 拡張文字 NEC 選定 IBM 拡張文字で定義される特殊文字
kgreek	ギリシャ文字を判別するクラス <hr/> JIS X 0208 6/1 - 7/33、6/33 - 56
krussian	ロシア文字を判別するクラス <hr/> JIS X 0208 7/1 - 7/33、7/49 - 81
kunit	単位記号を判別するクラス <hr/> JIS X 0208 1/75 - 83、2/82、83 JIS X 0212 2/80

表 2-5 日本語ローケル専用の文字分類クラス名とその定義範囲 続く

文字クラス名	意味 / 範囲 ¹
jsci	学術記号を判別するクラス <hr/> JIS X 0208 1/60 - 74, 2/26 - 33, 2/42 - 48 2/60 - 74
jgen	一般記号を判別するクラス <hr/> JIS X 0208 1/84 - 94, 2/1-14,2/84-89, 94 JIS X 0212 2/35, 75, 2/79-81 JIS X 0212 2/35, 75, 2/79 - 81
jpunct	一般記号を判別するクラス <hr/> JIS X 0208 1/2 - 37 JIS X 0212 2/34, 36

1. JIS X 0201 機能キャラクタ集合、ローマ文字用図形キャラクタ集合、片仮名用図形キャラクタ集合での XX/YY は XX 列 YY 行を意味します。JIS X 0208、JIS X 0212 での XX/YY は XX 区 YY 点を意味します。
2. JIS X 0212 の文字に関しては、ja および ja_JP.UTF-8 ローケルのみ該当します。

文字の照合順序 (LC_COLLATE カテゴリ)

日本語文字の照合順序は、文字コード値の順になります。ただし、ja_JP.PCK ローケルの場合、JIS X 0201 片仮名用図形キャラクタ集合は、JIS X 0201 ローマ文字用図形

キャラクタ集合、抹消文字の次になるように修正されています。文字の照合順序は、LC_COLLATE カテゴリの値に影響を受けます。

単語処理

JFP の提供する単語処理用共有オブジェクトは、単語処理用文字分類情報、詰め込み処理用単語分割・結合処理情報などをアプリケーションに供給します。これらの情報を利用するための API は今のところ公開されていません。

日付と時刻の表示形式 (LC_TIME カテゴリ)

SunOS および JFP における日本語の日付と時刻の表示形式は、以下のとおりです。

表 2-6 日付と時刻の表示形式

項目 (strftime(3C) での変換指定)	SunOS /JFP での定義
月の名前の省略形 (%b)	1月
	2月
	3月
	4月
	5月
	6月
	7月
	8月
	9月
	10月
	11月
	12月

表 2-6 日付と時刻の表示形式 続く

項目 (strftime(3C) での変換指定)	SunOS /JFP での定義
月の名前 (%B)	1月
	2月
	3月
	4月
	5月
	6月
	7月
	8月
	9月
	10月
	11月
	12月
曜日の名前の省略形 (%a)	(日)
	(月)
	(火)
	(水)
	(木)
	(金)
	(土)
曜日の名前 (%A)	日曜日
	月曜日

表 2-6 日付と時刻の表示形式 続く

項目 (strftime(3C) での変換指定)	SunOS /JFP での定義	
	火曜日	
	水曜日	
	木曜日	
	金曜日	
	土曜日	
時刻 (%X)	%H時%M分%S秒	
日付 (%x)	%Y年%m月%d日	
日付/時刻 (%c)	%Y年%m月%d日 %a %H時%M分%S秒	
午前/午後 (%p)	午前	
	午後	
午前/午後を伴う 12 時間表示形式(%r)	%p%I時%M分%S秒	
時刻の代替表示形式 (%EX)	%H時%M分%S秒	
日付の代替表示形式 (%Ex)	%EY%m月%d日	
日付および時刻の代替形式 (%Ec)	%EY%m月%d日 %a %H時%M分%S秒	
代替年表示 (%EY)		
	1927/01/01 - 1989/01/07	昭和%Ey年
	1989/01/08 - 1989/12/31	平成元年
1990/01/01 -	平成%Ey年	
日付/時刻、長形式 (%C)	%Y年%m月%d日 %a %H時%M分%S秒 %Z	

日本語対応 SunOS のユーザー環境設定

この章では、JFP によって日本語のサポートが可能になった SunOS に対して、必要に応じて行うさまざまな環境設定について説明します。ここでは端末エミュレータなどの CUI 環境の設定方法について説明しますので、GUI に関しては、第 7 章及び第 8 章を参照してください。

環境変数によるロケールの設定

国際化されたシステムプログラムおよびアプリケーションは、一般に、ユーザーが特定の環境変数をどのように設定しているかによって、どの言語および地域 (ロケール) を想定して動作するかを決定します。日本語環境を想定して動作を行うもっとも簡単な方法は、LANG という環境変数に ja (日本語 EUC を扱う)、ja_JP.PCK (PCK を扱う) または ja_JP.UTF-8 (UTF-8 を扱う) ロケール名を設定することです。環境変数の設定方法は、ユーザーの使用しているシェルプログラムによって異なりますが、csh (C シェル) の場合、以下のように実行します。

```
sun% setenv LANG locale1
```

常に日本語環境を使用するユーザーは、ホームディレクトリの初期化ファイル .login の中でこの設定を行うとよいでしょう。設定方法の例については第 5 章で説明します。sh (Bourne シェル) または ksh (Korn シェル) の場合には、以下のよう
に実行します。

```
sun$ LANG=locale1
sun$ export LANG
```

1. *locale* には ja、ja_JP.PCK または ja_JP.UTF-8 を指定します。

常に日本語環境を使用するユーザーは、ホームディレクトリの初期化ファイル `.profile` の中でこの設定を行うとよいでしょう。設定方法の例については第 5 章で説明します。

上に述べた設定によって、ユーザーが実行する国際化プログラムは日本語環境を想定して動作するようになり、一般に、メッセージや時刻の表示が日本語で行われたり、日本語文字の処理などが正しく行われたりするようになります。ただし、国際化されたウィンドウシステムが動いていない Sun ワークステーションの画面上など、英語は表示できても日本語を表示できないような状態では、日本語の文字は判読できません。このような場合には、一時的にロケールを C に設定し、英語のメッセージを表示させたほうがよいでしょう。この場合、`csh` (C シェル) では以下のように実行します。

```
sun% setenv LANG C
```

`sh` (Bourne シェル) または `ksh` (Korn シェル) では、以下のように実行します。

```
sun$ LANG=C
sun$ export LANG
```

ロケールの設定は、`LC_CTYPE`、`LC_NUMERIC`、`LC_TIME`、`LC_COLLATE`、`LC_MONETARY`、`LC_MESSAGES`、`LC_ALL` など、`LANG` よりも細かいカテゴリごとに行うこともできます。各ロケールカテゴリがプログラムの動作に与える影響については、各プログラムのリファレンスマニュアルを参照してください。

端末の設定

国際化された端末エミュレータウィンドウ (`dtterm`、`shelltool`、`cmdtool` など) や日本語端末上で正しく日本語の入出力を行うためには、`tty` (端末ドライバ) が日本語の文字データを正しく透過させ、日本語の文字幅に応じた処理を正しく行わなくてはなりません。このためには、次のようにロケールの設定が完了している状態で `setterm(1)` コマンドを実行して初期化する必要があります。

csch の場合 :

```
sun% setenv LANG locale1  
sun% setterm -x Codeset2
```

1. *locale* には ja、ja_JP.PCK または ja_JP.UTF-8 を指定します。

2. *Codeset* には JapanEUC、PCK または UTF-8 を指定します。

sh または ksh の場合 :

```
sun$ LANG= locale1; export LANG  
sun$ setterm -x Codeset2
```

1. *locale* には ja、ja_JP.PCK または ja_JP.UTF-8 を指定します。

2. *Codeset* には JapanEUC、PCK または UTF-8 を指定します。

注 -

日本語ロケール間でロケールを切り替える場合は、その都度、端末設定も行うようにしてください。使用する端末が同じでも、ロケールが異なると使用する STREAM モジュールが異なりますので、再度 `setterm` が実行されないと正しく表示できなくなります。

常に日本語端末環境を使用するユーザーは、ホームディレクトリの初期化ファイル (csch の場合には `.login` または `.cshrc`、sh の場合には `.profile`) の中でこの設定を行うとよいでしょう。次にロケールの指定と sh の対応表を示します。

表 3-1 ロケールの指定と sh の対応

ロケール	csh	sh、ksh
ja	% setenv LANG ja % setterm -x JapanEUC	% LANG=ja; export LANG % setterm -x JapanEUC
ja_JP.PCK	% setenv LANG ja_JP.PCK % setterm -x PCK	% LANG=ja_JP.PCK; export LANG % setterm -x PCK
ja_JP.UTF-8	% setenv LANG ja_JP.UTF-8 % setterm -x UTF-8	% LANG=ja_JP.UTF-8; export LANG % setterm -x UTF-8

端末オプションの詳細は、第 6 章および `setterm(1)` のマニュアルページを参照してください。

コマンド検索パスの設定

Solaris 2.6 オペレーティング環境およびその互換バージョン自体、およびそこに JFP がインストールされたシステムでは、同名のシステムコマンドがいくつか存在することがあります。たとえば、以下のコマンドがあります。

ls

<code>/usr/bin/ls</code>	国際化機能を持つ標準の <code>ls</code> コマンド
<code>/usr/ucb/ls</code>	BSD 互換性パッケージの <code>ls</code> コマンド
<code>/usr/xpg4/bin/ls</code>	XPG4 準拠の <code>ls</code> コマンド

mailx

```
/usr/bin/mailx
```

標準の mailx コマンド

```
/usr/SUNWale/bin/mailx
```

国際化機能の拡張された mailx コマンド

実行するコマンドのパスによっては、思い通りに国際化機能が利用できないことがあります。日本語環境で国際化機能を最大限に利用したいユーザーは、シェルのコマンド検索ディレクトリの優先順位を次のように設定するとよいでしょう。なお、対応する `/usr/xpg4/bin/<コマンド名>` がないときは `/usr/bin/<コマンド名>` が XPG4 準拠になっています。

```
/usr/xpg4/bin > /usr/SUNWale/bin > /usr/bin > /usr/ucb
```

設定方法は、ユーザーの使用しているシェルスプログラムによって異なりますが、`csch` (C シェル) の場合、以下のように `set` コマンドを実行します。

```
sun% set path = (/usr/xpg4/bin /usr/SUNWale/bin /usr/bin /usr/ucb .)
```

常に日本語環境を使用するユーザーは、ホームディレクトリの初期化ファイル `.cshrc` の中でこの設定を行うとよいでしょう。また、`.cshrc` ファイルを編集して `path` パラメータを変更した後、すぐにその内容を有効にしたいときには次のように `source` コマンドを実行します。

```
sun% source .cshrc
```

`sh` (Bourne シェル) または `ksh` (Korn シェル) の場合には、以下のようにパラメータ `PATH` を設定します。

```
sun$ PATH=/usr/xpg4/bin:/usr/SUNWale/bin:/usr/bin:/usr/ucb:.  
sun$ export PATH
```

常に日本語環境を使用するユーザーは、ホームディレクトリの初期化ファイル `.profile` の中でこの設定を行うとよいでしょう。

現在デフォルトで実行されるコマンドがどのディレクトリにあるかは、`which(1)` コマンドで確かめることができます。

オンラインマニュアル検索パスの設定

前の項で述べたように、Solaris 2.6 オペレーティング環境およびその互換バージョン自体、およびそこに JFP がインストールされたシステムでは、同名のシステムコマンドがいくつか存在することがあります。これに伴って、オンラインマニュアルページについても、同名のエントリが複数の場所に存在することがあります。たとえば、以下のマニュアルページがあります。

ls

<code>/usr/share/man/man1/ls.1</code>	国際化された標準の ls コマンド用
<code>/usr/share/man/man1b/ls.1b</code>	BSD 互換性パッケージの ls コマンド用

mailx

<code>/usr/share/man/man1/mailx.1</code>	標準の mailx コマンド用
<code>/usr/SUNWale/share/man/man1/mailx.1</code>	国際化された mailx コマンド用

参照するマニュアルページのパスによっては、思い通りに国際化機能の情報が得られないことがあります。日本語環境で国際化機能を最大限に利用するユーザーは、シェルのコマンド検索ディレクトリの優先順位とともにオンラインマニュアルページの検索パスの優先順位を次のように設定するとよいでしょう。

```
/usr/SUNWale/share/man > /usr/share/man
```

設定方法は、ユーザーの使用しているシェルプログラムによって異なります。

csh (C シェル) の場合は、次のように環境変数 MANPATH を設定します。

```
sun% setenv MANPATH /usr/SUNWale/share/man:/usr/share/man
```

常に日本語環境を使用するユーザーは、ホームディレクトリの初期化ファイル `.cshrc` でこの設定を行うとよいでしょう。

sh (Bourne シェル) または ksh (Korn シェル) の場合は、環境変数 MANPATH を次のように設定します。

```
sun$ MANPATH=/usr/SUNWale/share/man:/usr/share/man
sun$ export MANPATH
```

常に日本語環境を使用するユーザーは、ホームディレクトリの初期化ファイル `.profile` の中でこの設定を行うとよいでしょう。

初期化ファイル利用のヒント

ここではユーザーのホームディレクトリの `.login`、`.cshrc`、`.profile` といったシェルの初期化ファイルを利用して日本語環境用の設定を行う例を説明します。初期化ファイルの内容は個々のユーザーの必要に応じて変わるものであり、目的の達成方法も一つではありません。ここに挙げる例はあくまで参考であることに注意してください。

注 - `dtlogin` からログインした場合、以降に述べる (`.cshrc` を除く) 初期化ファイルは読み込まれないため、Solaris CDE アプリケーションには反映されません。

Solaris CDE アプリケーションの実行環境に反映させたい環境変数などの設定は、第7章を参照してください。

`.login`

ユーザーのホームディレクトリの `.login` ファイルは、ユーザーが C シェル (`csh`) を使ってシステムに `login` したときに C シェルの初期化に使われます (詳細は、`csh(1)` を参照してください)。一般に環境変数の設定などは、このファイルで行います。

```
# The initial user defaults for csh.

setenv LANG ja

setterm -x JapanEUC

if ( ! $?MANPATH ) then
    setenv MANPATH /usr/SUNWale/share/man:/usr/share/man
endif
```

最初の実行文で環境変数 `LANG` が、日本語ロケール名「`ja`」に設定されます。

"setterm -x JapanEUC" では、端末は日本語 EUC であると設定しています。

次の if 文は、オンラインマニュアルページの検索パスを条件に応じて設定します。この設定内容は、インストール時のパス設定、ソフトウェア製品の追加などに応じて変更が必要な場合があります。

.cshrc

ユーザーのホームディレクトリの .cshrc ファイルは、ユーザーが C シェル (csh) を起動したときに C シェルの初期化に使われます。C シェルがログインシェルとして起動されたときにも .cshrc ファイルは .login に先立って実行されます (詳細は、csh(1) を参照してください)。

```
set path=(/usr/xpg4/bin /usr/SUNWale/bin /usr/bin /usr/ucb .)
if ($?prompt) setterm -x JapanEUC
```

この例では、はじめの行でシェル変数 path に コマンド検索パスの設定だけを行なっています。この設定内容は、インストール時のパス設定、ソフトウェア製品の追加などに応じて変更が必要な場合があります。

次の行では、ロケールの設定を変えた後に C シェルが起動し直される場合などのため、"setterm -x JapanEUC" を実行しています。行の最初の if 文は、C シェルが端末以外の装置を標準入力として起動された場合に、stty が実行されてエラーメッセージが表示されるのを防ぐためのものですが、完全な解決策ではありません。

.profile

ユーザーのホームディレクトリの .profile ファイルは、ユーザーが Bourne シェル (sh) または Korn シェル (ksh) を使ってシステムにログインしたときに、シェルのユーザー固有の初期化に使われます (詳細は、sh(1) を参照してください)。


```

# The initial user defaults for sh.
LANG=ja
export LANG

PATH=/usr/xpg4/bin:/usr/SUNWale/bin:/usr/bin:/usr/ucb:.
export PATH

setterm -x JapanEUC

if [ "$MANPATH" = "" ]
then
    MANPATH=/usr/SUNWale/share/man:/usr/share/man
    export MANPATH
fi
ENV=$HOME/.kshrc
export ENV

```

この例では、環境変数 LANG でのロケール設定、環境変数 PATH でのコマンド検索パスの設定、tty のオプション設定が順次行われます。

"setterm -x JapanEUC" では、端末は日本語 EUC であると設定しています。

if 文では、それぞれ条件に応じて環境変数 MANPATH が設定されます。この設定内容は、インストール時のパス設定、ソフトウェア製品の追加などに応じて変更が必要な場合があります。

最後のパラメタ ENV は、Korn シェル (ksh) に対して起動時コマンド実行ファイル名を設定します。このパラメタ設定は、Bourne シェル (sh) に対しては特別な意味を持ちません。

Korn シェル 起動時コマンド実行ファイル

Korn シェル (ksh) に対して起動時コマンドファイルを用意する場合、そのファイル名をパラメタ ENV に設定します。パラメタ ENV の設定例については、32ページの「.profile」を参照してください。Korn シェルがログインシェルとして起動されたときにも、Korn シェル起動時コマンドファイルは .profile のあとで実行されます (詳細は、ksh(1) のマニュアルページを参照してください)。

```

export PATH=/usr/xpg4/bin:/usr/SUNWale/bin:/usr/bin:/usr/ucb:.
if expr "$-" : "i" > /dev/null; then
setterm -x JapanEUC
fi

```

この例では、はじめの行でパラメタ `PATH` にコマンド検索パスの設定だけを行なっています。この設定内容は、インストール時のパス設定、ソフトウェア製品の追加などに応じて変更が必要な場合があります。

次の 3 行では、ロケールの設定を変えた後に Korn シェルを起動し直す場合などのため、"`setterm -x JapanEUC`" を実行しています。

`if` 文は、Korn シェルが端末以外の装置を標準入力として起動された場合に、`setterm` が実行されてエラーメッセージが表示されるのを防ぐためのものですが、完全な解決策ではありません。

日本語対応 SunOS のシステム環境設定

この章では、システムで定義済みの SunOS の日本語環境を、システム管理者が必要に応じて変更する場合の方法と注意事項を説明します。

システムロケールの設定

ユーザーが個々のロケールを設定する場合と同様に、そのシステムのロケールも設定できます。このシステムロケールは、ユーザーがロケールを明示的に設定しないときのデフォルトロケールとして使われたり、ロケール依存の動きをするデーモンのロケールとして使われたりします。

インストール時に「言語とロケールの選択 (Select Language and Locale)」画面で、選択したロケールがシステムロケールとして設定されています。システムロケールは以下のファイルで設定していますので、このファイルの LANG の値を変更することによって、システムロケールを変更できます。詳細は、init(1M) を参照してください。

```
sun% more /etc/default/init
:
TZ=Japan
LANG=ja
```

電子メールの送受信に関する設定

in.comsat の設定

JFP では、複数バイト言語国際化のために拡張された in.comsat (/usr/SUNWale/sbin/in.comsat) を提供しています。第 5 章の 81 ページの「メール到着の通知」で到着した日本語メールを、Solaris CDE や OpenWindows のコンソールウィンドウなどでコード変換して見るためには、以下のように install_comsat を実行する必要があります。

```
sun# /usr/SUNWale/sbin/install_comsat ja
```

詳細は、install_comsat(1M) および inetd(1M) のマニュアルページを参照してください。

日本語プリンタに関する設定

JFP では、LP 印刷サービスを使用して、ドットマトリックス漢字プリンタ (EPSON VP-5085、NEC PC-PR201)、日本語ページプリンタ (Canon LASERSHOT)、日本語 PostScript プリンタで日本語テキストを印刷するために、これらの変換フィルタ群を提供します。

日本語 dumb プリンタのサポート

JFP では、LP 印刷サービスを使用して、ドットマトリックス漢字プリンタ (EPSON VP-5085、NEC PC-PR201)、日本語ページプリンタ (Canon LASERSHOT) で日本語テキストを印刷できます。

EPSON VP-5085 用のデータベースは EPSON ESC/P の ESC/P24-J84 に準拠したコントロールコード、NEC PR201 用のデータベースは NEC 201PL に準拠したコントロールコードを使用しています。そのため JFP では、これらのコントロールコードをサポートする他のプリンタでも日本語テキストを印刷できます。

提供されるファイル

JFP では、日本語 dumb プリンタをサポートするために以下のファイルを提供します。

表 4-1 日本語 dumb プリンタのサポート用に提供されるファイル

プロセッサの種類	キーボードの種類
<code>/usr/share/lib/terminfo/c/ canon-ls-a408</code>	Canon LASERSHOT A408 用の terminfo データベース (Canon LASERSHOT の LIPS に準拠)
<code>/usr/share/lib/terminfo/e/ epson-vp5085</code>	EPSON-VP5085 用の terminfo データ ベース (EPSON ESC/P の ESC/P24-J84 に準拠)
<code>/usr/share/lib/terminfo/n/nec-pr201</code>	NEC PR201 用の terminfo データベ ース (NEC 201PL に準拠)
<code>/usr/lib/lp/text/jprconv</code>	日本語テキスト印刷フィルタ
<code>/usr/lib/lp/text/jprconv.conf</code>	日本語テキスト印刷フィルタ用設定 ファイル
<code>/etc/lp/fd/jprconv.fd</code>	日本語テキスト印刷フィルタ記述子 ファイル
<code>/usr/lib/lp/model/jstandard</code>	日本語プリンタ・インタフェースプログ ラム

lp のプリンタ固有オプション

日本語プリンタ・インタフェースプログラムでは、標準のインタフェースプログラムに表 4-2 に示すプリンタ固有のオプションが追加されています。その他のオプションは標準インタフェースプログラムと同じです。詳細は、lp(1) のマニュアルページを参照してください。

表 4-2 lp のプリンタ固有オプション

オプション	内容
<code>x=cols</code>	左マージンを <code>cols</code> カラムに設定する
<code>landscape</code>	用紙を横長にして印刷する (Canon LASERSHOT のみ有効)

プリンタ固有のオプションの使用例を以下に示します。

```
sun% lp -o x=30 textfile.txt
sun% pr -2 -l46 -w170 -t textfile.txt | lp -o nobanner -o landscape
```

jprconv フィルタ固有のオプション

以下のフィルタ固有のオプションが `lp` を通して利用可能です。

表 4-3 jprconv フィルタ固有のオプション

オプション	内容
<code>-y PCK</code>	指定したファイルの文字コードを PCK に指定
<code>-y eucJP</code>	指定したファイルの文字コードを eucJP に指定
<code>-y UTF-8</code>	指定したファイルの文字コードを UTF-8 に指定

フィルタ固有のオプションの使用例を以下に示します。

```
sun% lp -y PCK textfile.pck
```

制限事項

日本語テキスト印刷フィルタを使用する場合、次の制限事項があります。

制限事項

- PostScript ファイルを印刷できない

- lp のプリンタ固有オプションで、length、width、lpi、および cpi はサポートされない
- UTF-8 ではユーザー定義文字を除いて日本語 EUC で表現できる文字のみ印刷可能です。

日本語 PostScript プリンタのサポート

JFP では、日本語 PostScript プリンタで日本語テキストを印刷できます。日本語 PostScript プリンタに日本語テキストを印刷する方法は、jtops(1) を用いてクライアント側で PostScript に変換して、プリンタサーバーに渡す方法とサーバー側で変換する方法があります。ここでは、サーバー側で変換する方法を使用する場合のプリンタの設定方法について説明します。

提供されるファイル

JFP では、日本語 PostScript プリンタをサポートするために以下のファイルを提供します。

表 4-4 日本語 PostScript プリンタのサポート用に提供されるファイル

<code>/usr/share/lib/terminfo/c/canon-ls-a408</code>	Canon LASERSHOT A408 用の terminfo データベース (Canon LASERSHOT の LIPS に準拠)
<code>/usr/lib/lp/postscript/jpostprint</code>	日本語テキストを日本語 PostScript に変換するフィルタ
<code>/usr/lib/lp/postscript/jpostprint.conf</code>	上記フィルタの設定ファイル
<code>/etc/lp/fd/jpostprint.fd</code>	上記フィルタの記述子ファイル

jpostprint フィルタ固有のオプション

以下のフィルタ固有のオプションが lp を通して利用可能です。

表 4-5 jpostprint フィルタ固有のオプション

オプション	内容
<code>-o length=len</code>	1 ページの行数を <i>len</i> に指定
<code>-P page-list</code>	<i>page-list</i> で指定するページ範囲を印刷
<code>-y group</code>	1 ページあたり 2 個の論理ページで印刷
<code>-y group=n</code>	1 ページあたり <i>n</i> 個の論理ページで印刷
<code>-y portrait</code>	縦長で印刷
<code>-y landscape</code>	横長で印刷
<code>-y magnify=n</code>	<i>n</i> 倍に拡大して印刷
<code>-y PCK</code>	指定したファイルの文字コードを PCK に指定
<code>-y eucJP</code>	指定したファイルの文字コードを eucJP に指定
<code>-y UTF-8</code>	指定したファイルの文字コードを UTF-8 に指定

フィルタ固有のオプションの使用例を以下に示します。

```
sun% lp -P 2,3 -y PCK textfile.pck
```

制限事項

- UTF-8 ではユーザー定義文字を除いて日本語 EUC で表現できる文字のみ印刷可能です。

プリンタの追加方法

日本語テキストを印刷するには、プリンタを正しく追加する必要があります。この場合、ローカルプリンタとして追加するか、リモートプリンタとして追加するかによって追加方法が異なります。

まず、各プリンタに共通する手順を説明します。

注 - x86 上にプリンタデバイスが存在しない場合は、『Solaris 情報ライブラリ (Intel 版)』の「デバイスの構成」を参照してあらかじめプリンタデバイスを正しく作成しておく必要があります。

1. システムの電源を切ります。
2. システムとプリンタをプリンタケーブルで接続します。
3. システムの電源を入れ、**Solaris** を起動します。
4. スーパーユーザーになります。

注 - 一般ユーザーでシステム管理ツール (Admintool) を使用する際は、`sysadmin` グループ (GID 14) の一員になる必要があります。

5. まだ起動していなければ、**Solaris CDE** または **OpenWindows** を起動します。

次に、各プリンタに固有の手順について説明します。

ローカルプリンタとして追加する場合

Admintool を使用してローカルプリンタを追加する方法を説明します。

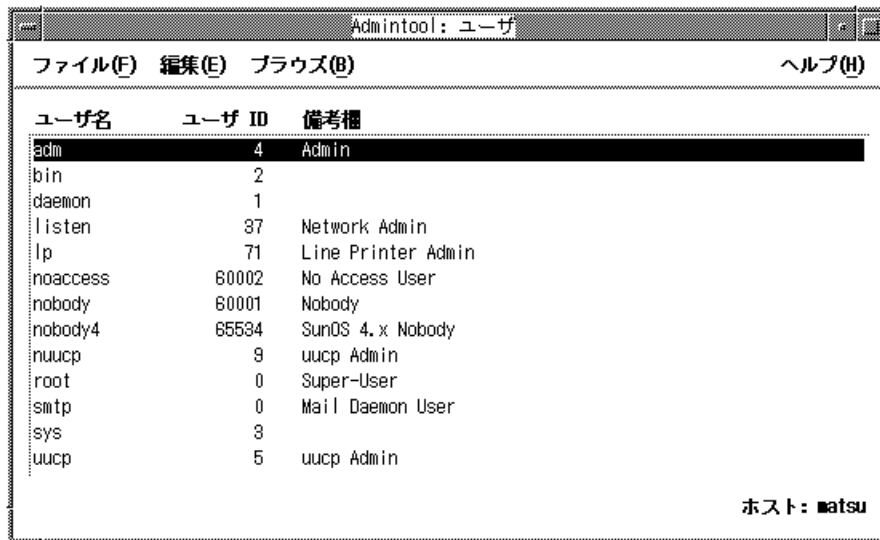
注 - Admintool の詳細は、『Solaris のシステム管理』を参照してください。

この説明では、プリンタ名を `vp5085`、プリンタタイプを `epson-vp5085`、デバイス名を `/dev/term/a` と仮定しています。

以下に手順を示します。

1. **Admintool** を起動します。

```
sun# admintool &
```

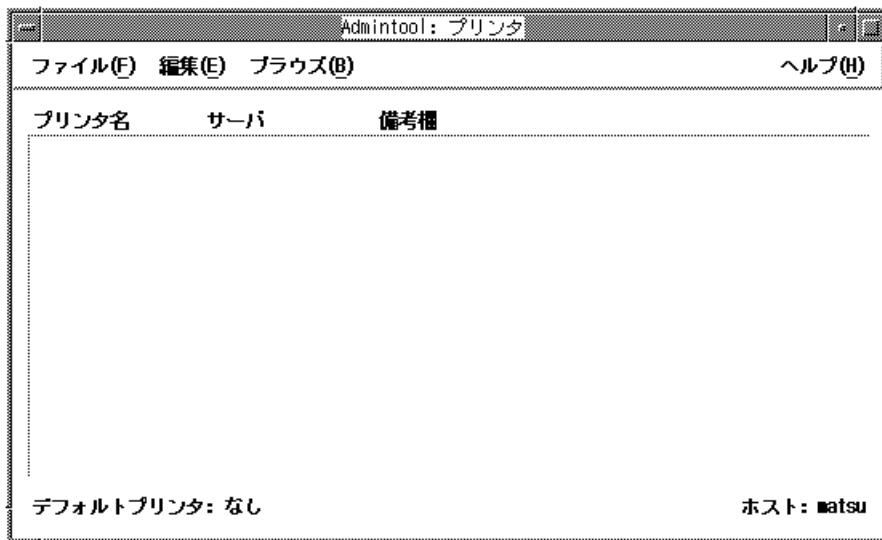


The screenshot shows a window titled "Admin tool: ユーザ" with a menu bar containing "ファイル(F)", "編集(E)", "ブラウズ(B)", and "ヘルプ(H)". The main area displays a table of users with columns for "ユーザ名", "ユーザ ID", and "備考欄". The "adm" user is highlighted. The bottom right corner of the window shows "ホスト: matsu".

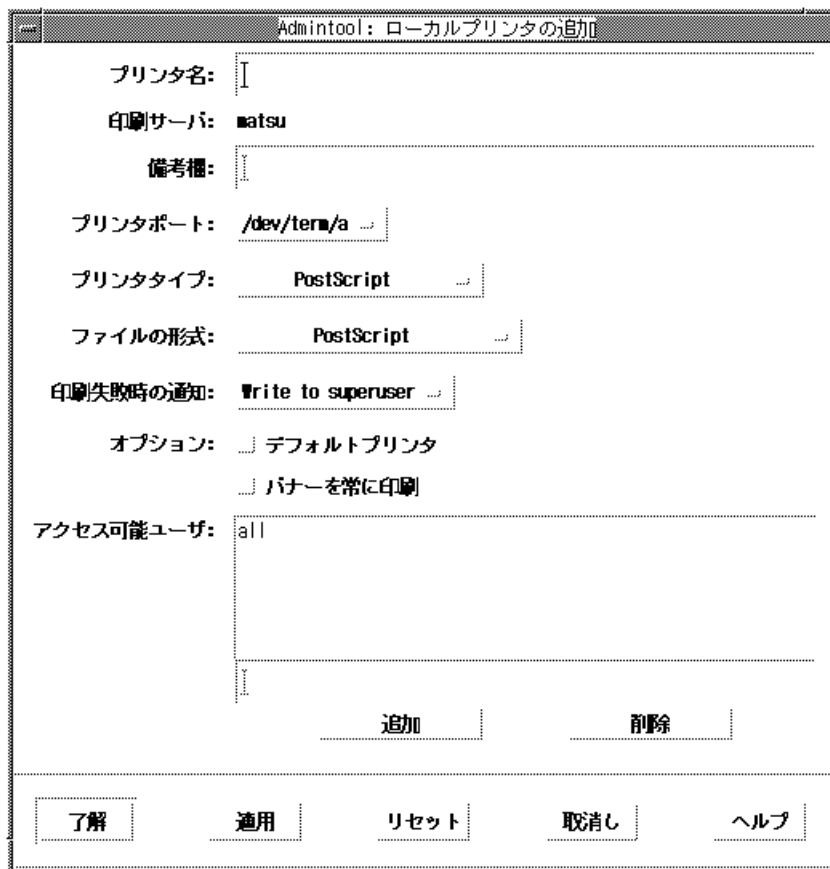
ユーザ名	ユーザ ID	備考欄
adm	4	Admin
bin	2	
daemon	1	
listen	37	Network Admin
lp	71	Line Printer Admin
noaccess	80002	No Access User
nobody	80001	Nobody
nobody4	85534	SunOS 4.x Nobody
nuucp	9	uucp Admin
root	0	Super-User
smtp	0	Mail Daemon User
sys	3	
uucp	5	uucp Admin

ホスト: matsu

2. 「ブラウズ」上をクリックして、「プリンタ」を選択します。



3. 「編集」上をクリックして「追加」を選択し、さらに「ローカルプリンタ...」を選択します。
以下のウィンドウが表示されます。



4. 「プリンタ名」テキストフィールドに、ローカルプリンタ名を入力します (この例では **vp5085**)。
5. 「プリンタポート」上でマウスのメニューボタンを押し、該当するデバイスを選択します (この例では /dev/term/a)。
6. 「プリンタタイプ」上でマウスのメニューボタンを押し、「その他...」を選択します。そして、表示されるポップアップウィンドウに、以下の表から該当するプリンタタイプを入力し、「了解」ボタンを押します (この例では、**epson-vp5085**)。

表 4-6 プリンタタイプ

コントロールコード	プリンタタイプ
Canon LASERSHOT	canon-ls-a408
EPSON ESC/P	epson-vp5085
NEC 201PL	nec-pr201
日本語 PostScript	PostScript

7. 「ファイルの形式」上でマウスのメニューボタンを押し、「**None**」を選択します。
8. プリンタを、デフォルトプリンタとしてシステムに追加したい場合は、オプションの「デフォルトプリンタ」のチェックボックスをチェックします。
9. 「了解」ボタンをクリックします。

Admintool: ローカルプリンタの追加

プリンタ名: vp5085

印刷サーバ: matsu

備考欄:

プリンタポート: /dev/term/a

プリンタタイプ: epson-vp5085

ファイルの形式: None

印刷失敗時の通知: Write to superuser

オプション:
 デフォルトプリンタ
 パナーを常に印刷

アクセス可能ユーザ: all

追加 削除

了解 適用 リセット 取消し ヘルプ

10. 次のように入力して、印刷フィルタを登録します。

日本語 dumb プリンタの場合 :

```
sun# cd /etc/lp/fd
sun# lpfilter -f jprconv -F jprconv.fd
```

日本語 PostScript プリンタの場合 :

```
sun# cd /etc/lp/fd
sun# lpfilter -x -f postprint
sun# lpfilter -f jpostprint -F jpostprint.fd
```

11. 必要に応じて、インタフェースプログラムの構成を変更します。

常に、バナーページを印刷しないなどの場合は、インタフェースプログラムを次のように編集してください。

```
sun# cd /usr/lib/lp/model
sun# vi jstandard
```

"Default nobanner" の "nobanner="no" の部分を「nobanner="yes"」に変更します。

```
:
#
# Default nobanner
# nobanner="no" : banner page will be printed as default
# nobanner="yes" : banner page will not be printed as default
#
nobanner="yes"
:
```

12. 次のように入力してインタフェースプログラムを登録します。

-p の後には、44ページの手順4のプリンタ名の設定で入力したプリンタ名を入力します (この例では vp5085)。

```
sun# lpadmin -p vp5085 -i /usr/lib/lp/model/jstandard
```

13. 44ページの手順5の印刷ポートの設定でデバイスとして /dev/bpp0 以外を指定した場合は、次のように入力して、stty オプションを設定します。

```
sun# lpadmin -p vp5085 -o "stty=-opost"
```

14. システムをリブートします。

```
sun# sync;sync;sync;reboot
```

リモートプリンタとして追加する場合

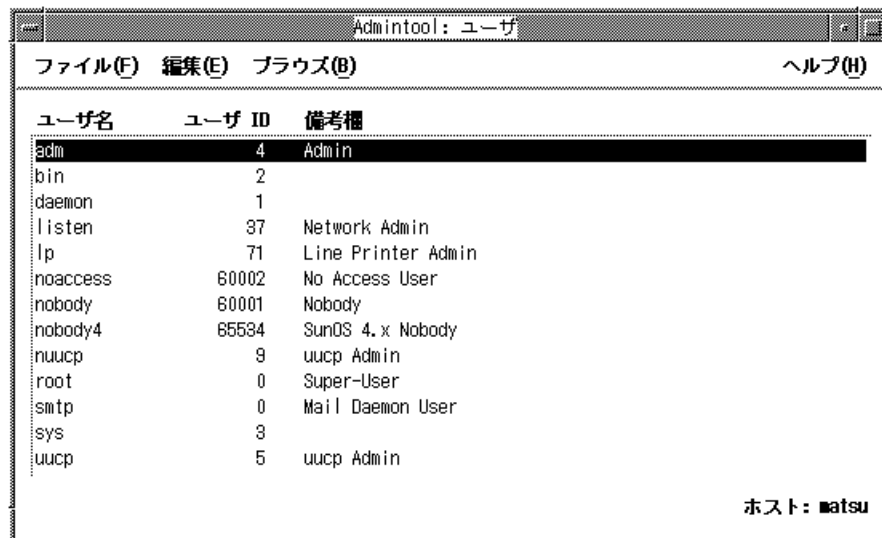
ここでは、Admintool を使用してリモートプリンタを追加する方法を説明します。

注 - Admintool の詳細は、『Solaris のシステム管理』を参照してください。

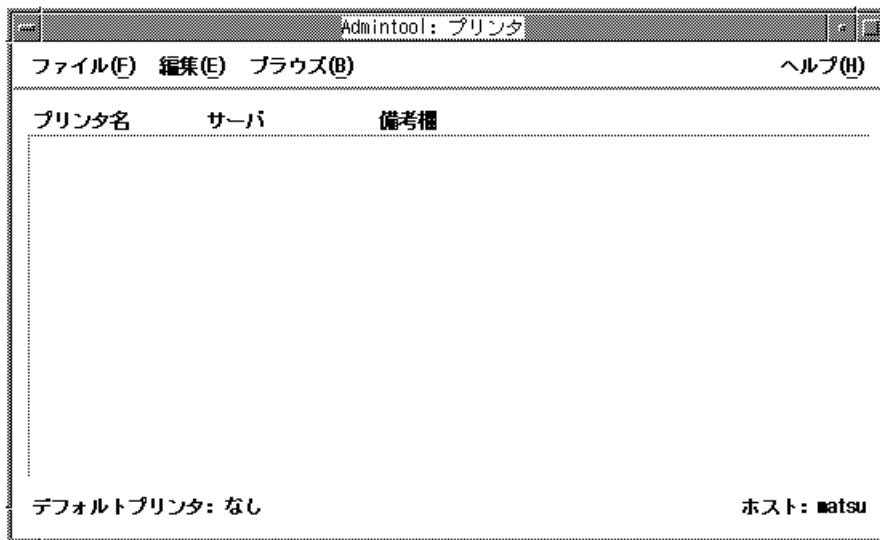
以下に手順を示します。

1. Admintool を起動します。

```
sun# admintool &
```



2. 「ブラウズ」上をクリックして、「プリンタ」を選択します。



- 「編集」上をクリックして「追加」を選択し、さらに「プリンタへのアクセス...」を選択します。
以下のウィンドウが表示されます。



4. 「プリンタ名」テキストフィールドに、リモートプリンタ名を入力します (この例では **vp5085**)。
5. 「印刷サーバ」テキストフィールドに、リモートプリンタを接続するシステム名を入力します (この例では **hiraki**)。
6. プリンタをシステムのデフォルトプリンタにしたい場合は、オプションの「デフォルトプリンタ」のチェックボックスをチェックします。
7. 「了解」ボタンをクリックします。

Admintool: プリンタへのアクセス

プリンタクライアント: **matsu**

プリンタ名: vp5085

印刷サーバ: hiraki

備考欄: |

オプション: デフォルトプリンタ

了解 適用 リセット 取消し ヘルプ

8. `printers.conf` ファイルを編集します。

```
sun# cd /etc
sun# vi printers.conf
```

`vi` などで `printers.conf` ファイルの該当するサーバー部分 `":bsdaddr=hiraki,sparc140:"` に、以下のように「Solaris」を追加します。

```
:bsdaddr=hiraki,sparc140,Solaris:
```

以上の手順が終了すると、lp コマンドを使用して、追加したリモートプリンタに出力できます。ここで設定したプリンタの利用法に関しては、第 6 章を参照してください。

JIS X 0212、ユーザー定義文字、ベンダー定義文字の印刷サポート

JFP では、日本語 dumb プリンタや日本語 PostScript プリンタでの、JIS X 0212、ユーザー定義文字、およびベンダー定義文字を含む日本語テキストの印刷をサポートしています。これらの文字を印刷するためには、そのフォントをプリンタサーバー上に正しくインストールする必要があります。ここでは、これらのフォントのインストール方法について説明します。

ユーザー定義文字とベンダー定義文字の定義領域に関しては、第 2 章を参照してください。また、ユーザー定義文字の作成方法は、第 14 章を参照してください。

日本語 dumb プリンタで使用する場合

ユーザー定義文字のフォントは sdtudctool (1) で出力される PCF フォント、JIS X 0212 とベンダー定義文字のフォントは SUNWjxcft に含まれる 24 ドットの PCF フォントだけが使用できます。

フォントのインストール方法

プリンタサーバー上で以下を実行します。

■ JIS X 0212 とベンダー定義文字フォントのインストール

1. **SUNWjxcft** パッケージがインストールされているかどうかを確認します。

```
sun# pkginfo SUNWjxcft
```

2. **SUNWjxcft** パッケージがインストールされている場合は、次のようなメッセージが表示されます。

```
sun# SUNWjxcft Japanese X Window System common (not required) fonts
```

3. ここで **SUNWjxcft** が見つからなかった場合は、次のコマンドを実行します。

```
sun# pkgadd -d <Solaris CD>/Product SUNWjxcft
```

上記 <Solaris CD> は、Solaris CD のインストールイメージがあるディレクトリを指定します。

■ ユーザー定義文字フォントのインストール

1. sdtudctool でユーザー定義文字フォントを作成します。
2. 作成したフォントをインストールします。

```
sun% cd $HOME/.Xlocale/$LANG/fonts/UDC/Bitmaps
sun% su
sun# mkdir -p /usr/openwin/lib/locale/$LANG/X11/fonts/UDC/Bitmaps
sun# cp UDC24.pcf /usr/openwin/lib/locale/$LANG/X11/fonts/UDC/Bitmaps
```

日本語 **PostScript** プリンタで使用する場合

ユーザー定義文字のフォントは sdtudctool (1) で出力される Type1 フォント、JIS X 0212 とベンダー定義文字のフォントは SUNWjcs3f に含まれる Type1 フォントだけが使用できます。

フォントのインストール方法

プリンタサーバー上で以下を実行します。

■ JIS X 0212 とベンダー定義文字フォントのインストール

1. **SUNWjcs3f** パッケージがインストールされているかどうかを確認します。

```
sun# pkginfo SUNWjcs3f
```

2. **SUNWjcs3f** パッケージがインストールされている場合は次のようなメッセージが表示されます。

```
sun# SUNWjcs3f Japanese JIS X0212 Type1 fonts for printing
```

3. ここで **SUNWjcs3f** が見つからなかった場合は、次のコマンドを実行します。

```
sun# pkgadd -d <Solaris CD>/Product SUNWjcs
```

上記 <Solaris CD> は、Solaris CD のインストールイメージがあるディレクトリを指定します。

■ ユーザー定義文字フォントのインストール

1. sdtudctool でユーザー定義文字フォントを作成します。
2. 作成したフォントをインストールします。

```
sun% cd $HOME/.Xlocale/$LANG/fonts/UDC/Type1
sun% su
sun# mkdir -p /usr/openwin/lib/locale/$LANG/X11/fonts/UDC/Type1
sun# cp UDC*.pfa /usr/openwin/lib/locale/$LANG/X11/fonts/UDC/Type1
```

また、jpostprint を起動し、クライアント側で PostScript ファイルに変換することによりプリンタサーバーと異なった独自の外字の印刷することも可能です。

```
sun% /usr/lib/lp/postscript/jpostprint <テキストファイル名>
```

この場合は、\$HOME/.Xlocale/\$LANG/fonts/UDC/Type1 にある外字フォントが使用されます。

日本語キーボードのサポート

JFP では、次の日本語キーボードをサポートしています。

プロセッサの種類	キーボードの種類
SPARC	日本語タイプ 4 キーボード 日本語タイプ 5 キーボード
x86	106 キーボード J3100 キーボード

日本語タイプ 4、5 キーボード

日本語タイプ 5 キーボードの制限事項については、U.S. タイプ 5 キーボードと同じです。『Sun タイプ 5 キーボード プロダクト・ノート』を参照してください。

日本語キーボードは、SunOS の「各国語キーボードサポート」機能に基づき、日本語タイプ 4 キーボードの場合、自動的にレイアウトコード 0x20 のキーボードと判定され、対応するキーテーブルが読み込まれます。同様に日本語タイプ 5 キーボードの場合、自動的にレイアウトコード 0x31 のキーボードと判定され、対応するキーテーブルが読み込まれます。キーテーブルは /usr/share/lib/keytables に置かれ、ブート時に読み込まれます。キーテーブルの名称は、日本語タイプ 4 キーボードの場合は layout_20 で、japan と同じものです。同様に日本語タイプ 5 キーボードの場合は layout_31 で、japan_5 と同じものです。

国際化 SunOS コマンドの利用

SunOS 上の多くのコマンドは、複数バイト文字データを適切に処理できるように国際化されています。この章では、代表的なコマンドをいくつか取り上げ、日本語環境での利用方法について説明します。

この章で説明されていない事項およびコマンドについては、それぞれのコマンドのマニュアルページなどを参照してください。複数バイト言語を対象として国際化されているコマンドのマニュアルページでは一般に、ENVIRONMENT (環境) の項に LC_* 環境変数に応じた動作の変化が記述されています。

vi エディタ

vi は SunOS 上の標準的なテキストエディタです。ウィンドウ環境への依存性がなく、広範囲のファイルタイプに対応しているため、さまざまな端末で多目的に使いこなすことができます。

この節では、vi を例に SunOS で国際化されているコマンドの日本語環境での使用方法を説明します。vi の機能全般については、『OpenWindows ユーザーズガイド (上級編)』および vi (1) のマニュアルページなどを参照してください。また、実際の日本語入力方法については、『Wnn6 ユーザーズガイド』、『ATOK8 ユーザーズガイド』、または『cs00 ユーザーズガイド』を参照してください。

注 - この項の説明で "「xxx」を入力してください" という表現は、xxx が日本語の場合は日本語を入力し、無変換の状態に戻すまでを指します。

```
1492年、クリストファー・コロンブスは、スペイン女王イザベルの援助を受け、黄金に輝く地、Z I P A N G U（日本）を求めてスペインのパロスを出発した。3箇月後、西インド諸島サンサルバドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきっていた。
```

```
~  
~  
~  
~  
~  
~  
~  
~
```

```
:set showmode
```

挿入モード

上の例では、画面2行目の最後のカラムに「ド」の文字がかかりませんが行内に納まりません。このような場合には「ド」の文字は次の行に回され、2行目の行末の1カラムはスキップされ、それを示すために「~」の文字が表示されます。行末のカラムのスキップを示す文字は、環境変数 MC_FILLER を使って「~」から任意のASCII文字に変更できます。

注 - 本文では、現在のカーソル位置を全角文字の上にある場合は全角アンダースコア (_)、半角文字の上にある場合は半角アンダースコア () で表示していますが、これは使用する端末によって変わります。

たとえば、次のように設定します。

```
sun% setenv MC_FILLER ' '
```

この設定をしたあと、vi を立ち上げて同じテキストを入力すると、スキップを示す文字がスペースに変わります。

```
1492年、クリストファー・コロンブスは、スペイン女王イザベルの援助を受け、黄金に輝く地、Z I P A N G U（日本）を求めてスペインのパロスを出発した。3箇月後、西インド諸島サンサルバドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきっていた。
```

```
~  
~  
~  
~  
~  
~  
~  
~
```

```
:set showmode
```

挿入モード

次に「クリストファー」の最後の1文字「ー」を削除してみます。カーソルを削除したい「ー」に移動しxキーを入力すると、「ー」の1文字が削除され、画面は次のようになります。

```
1492年、クリストファ・コロンブスは、スペイン女王イザベルの援助を受け、黄金に輝く
地、Z I P A N G U (日本) を求めてスペインのパロスを出発した。3箇月後、西インド
諸島サンサルバドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じき
っていた。
~
~
~
~
~
~
~
~
~
~
:set showmode
```

さらに2行目の文字列、「Z I P A N G U」の「I P A N G U」を小文字に変換してみます。カーソルを「Z I P A N G U」の「I」に移動して「6~」と入力します。これにより、1文字を大文字・小文字変換して次の文字に移動する操作が6回繰り返され、画面は次のように変わります(~キーを6回入力しても同じです)。

```
1492年、クリストファ・コロンブスは、スペイン女王イザベルの援助を受け、黄金に輝く
地、Z i p a n g u_(日本) を求めてスペインのパロスを出発した。3箇月後、西インド
諸島サンサルバドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じき
っていた。
~
~
~
~
~
~
~
~
~
~
:set showmode
```

単語単位の日本語処理

SunOS上のいくつかのコマンドでは、単語間の区切りとして空白を使用しない言語用に拡張された、単語の識別処理がサポートされています。

viでは、コマンドモードでw(word)キーを入力することにより、カーソルが日本語の単語識別用文字分類に基づいて次の単語の先頭に移動します。同様に、b(back)


```
1492年、クリストファ・コロンブスは、スペイン女王イザベルの援助を受け、黄金に輝く
地、Zipang (日本) を求めてスペインのパロスを出発した。3箇月後、西インド
諸島サンサルバドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じき
っていた。
```

```
~
~
~
~
~
~
~
```

```
:set showmode
```

変更モード

「ジパング」と入力して Esc キーを入力すると、「Zipangu」が「ジパング」に置き換えられます。

```
1492年、クリストファ・コロンブスは、スペイン女王イザベルの援助を受け、黄金に輝く
地、ジパング (日本) を求めてスペインのパロスを出発した。3箇月後、西インド諸島サ
ンサルバドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきって
いた。
```

```
~
~
~
~
~
~
~
```

```
:set showmode
```

同様に、このような単位の単語を削除することもできます。カーソルキーを使ってカーソルを「イザベル」の「イ」に移動し dw と入力すると、カーソル位置から現在の単語の最後の「ル」までが削除されます。

```
1492年、クリストファ・コロンブスは、スペイン女王の援助を受け、黄金に輝く地、ジパ
ング (日本) を求めてスペインのパロスを出発した。3箇月後、西インド諸島サンサルバ
ドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきっていた。
```

```
~
~
~
~
~
~
~
```

```
:set showmode
```

改行マージンと連結コマンド (行の分割と連結)

入力テキストの改行は、Return キーを入力することによって任意に行えますが、vi の set コマンドと wm オプションを使用すると、行の長さを自動的にコントロールできます。さらに、日本語環境では原則として日本語文字 1 文字が改行の単位とみなされます。たとえば、次の例では画面の幅が ASCII の 80 カラム分なので、コマンドモードで「:set wm=58」を入力することにより、画面右側に 58 カラムのマージン (余白) が確保され、行の長さは $80-58=22$ カラム以下になるようにコントロールされます。

```
1492年、クリストファ・コロンブスは、スペイン女王の援助を受け、黄金に輝く地、ジパング（日本）を求めてスペインのパロスを出発した。3箇月後、西インド諸島サンサルバドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきっていた。
```

```
—
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
:set wm=58
```

コマンドモードで o (open) キーを入力してオープンモードに移り、入力を次のように続けます。

「コロンブスは、その後もジパングをめざし、数回の航海でキューバ島やイスパニオラ島を発見した。しかし、最後までジパングを見つけることはできなかった。」

Return キーを入力せずに続けて入力しても、テキストは自動的に次のように改行されます。

```
1492年、クリストファ・コロンブスは、スペイン女王の援助を受け、黄金に輝く地、ジパ  
ング（日本）を求めてスペインのパロスを出発した。3箇月後、西インド諸島サンサルバ  
ドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきっていた。
```

```
コロンブスは、その後も  
ジパングをめざし、数回  
の航海でキューバ島やイ  
スパニオラ島を発見し  
た。しかし、最後までジ  
パングを見つけることは  
できなかった。_
```

```
~
```

```
:set wm=58
```

```
オープンモード
```

ここで、「スパニオラ島を発見し」の行が次の行より1文字分短く改行されるのは、テキストを清書するときと同様に禁則処理が働いているためです。

次に行を連結してみましょう。Esc キーを入力してコマンドモードに移り、「コロンブス...」の行にカーソルを移動します。

```
1492年、クリストファ・コロンブスは、スペイン女王の援助を受け、黄金に輝く地、ジパ  
ング（日本）を求めてスペインのパロスを出発した。3箇月後、西インド諸島サンサルバ  
ドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきっていた。
```

```
コロンブスは、その後も  
ジパングをめざし、数回  
の航海でキューバ島やイ  
スパニオラ島を発見し  
た。しかし、最後までジ  
パングを見つけることは  
できなかった。
```

```
~
```

```
:set wm=58
```

ここで J (Shift-j) キーを入力すると、現在の「コロンブス...」の行に次の「ジパング...」の行が連結されます。


```
1492年、クリストファ・コロンブスは、スペイン女王の援助を受け、黄金に輝く地、ジパ  
ング（日本）を求めてスペインのパロスを出発した。3箇月後、西インド諸島サンサルバ  
ドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきっていた。
```

```
コロンブスは、その後もジパングをめざし、数回  
の航海でキューバ島やイ  
スパニオラ島を発見し  
た。しかし、最後までジ  
パングを見つけることは  
できなかった。  
~  
~  
:set wm=58
```

このように日本語文字の行の連結の場合には、英語の行の連結の場合とは違い、スペースが挿入されず、改行の場合と同様に自然な日本語処理が行えます。日本語固有のこれらの単語処理は、JFP の提供する各国語化によって実現されています。¹

日本語文字を含む文字列の検索と置換

いくつかの SunOS コマンドと同様、vi でも日本語文字を含む文字列パターンの処理がサポートされます。コマンドモードで「/」キーを入力すると画面の最下行に「/」が表われ、検索する文字列を待つ状態になります。ここで、検索する文字列として「コロンブス」と入力してみましょう。

```
1492年、クリストファ・コロンブスは、スペイン女王の援助を受け、黄金に輝く地、ジパ  
ング（日本）を求めてスペインのパロスを出発した。3箇月後、西インド諸島サンサルバ  
ドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきっていた。
```

```
コロンブスは、その後もジパングをめざし、数回  
の航海でキューバ島やイ  
スパニオラ島を発見し  
た。しかし、最後までジ  
パングを見つけることは  
できなかった。  
~  
~  
/コロンブス_
```

1. このような国際化された単語処理に慣じみのない既存の vi のユーザーは、これらの機能にとまどいを覚えるかも知れません。そのような場合には、環境変数 MB_CONSERVATIVE_EDIT を設定してください。テキストの清書に準じた日本語単語処理は vi に適用されなくなり、vi は日本語に対しても英語と同様の動作をするようになります。

Return キーを入力すると、カーソルは先頭行の「コロンブス」の先頭に移動します。

```
1492年、クリストファ・コロンブスは、スペイン女王の援助を受け、黄金に輝く地、ジパ  
ング（日本）を求めてスペインのパロスを出発した。3箇月後、西インド諸島サンサルバ  
ドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきっていた。
```

```
コロンブスは、その後もジパングをめざし、数回  
の航海でキューバ島やイ  
スパニオラ島を発見し  
た。しかし、最後までジ  
パングを見つけることは  
できなかった。  
~  
~  
/  
コロンブス
```

n (next) キーを入力すると、カーソルは画面 2 段落目の最初の行の「コロンブス」の先頭に移動します。

```
1492年、クリストファ・コロンブスは、スペイン女王の援助を受け、黄金に輝く地、ジパ  
ング（日本）を求めてスペインのパロスを出発した。3箇月後、西インド諸島サンサルバ  
ドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきっていた。
```

```
コロンブスは、その後もジパングをめざし、数回  
の航海でキューバ島やイ  
スパニオラ島を発見し  
た。しかし、最後までジ  
パングを見つけることは  
できなかった。  
~  
~  
/  
コロンブス
```

次に、1 行目の ASCII の英数字を含む「1492年」の文字列をコードセット 1 の「1492年」に変更してみます。正規表現を使って「1492年」を「...2年」で指定することにし、「:%s/...2年/1492年/g」を入力します。

```
1492年、クリストファ・コロンブスは、スペイン女王の援助を受け、黄金に輝く地、ジパング（日本）を求めてスペインのパロスを出発した。3箇月後、西インド諸島サンサルバドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきっていた。
```

```
コロンブスは、その後もジパングをめざし、数回の航海でキューバ島やイスパニオラ島を発見した。しかし、最後までジパングを見つけることはできなかった。
```

```
~  
~  
:%s/...2年/1492年/g_
```

Return キーを入力すると文字列が置き換えられ、画面は次のようになります。

```
1492年、クリストファ・コロンブスは、スペイン女王の援助を受け、黄金に輝く地、ジパング（日本）を求めてスペインのパロスを出発した。3箇月後、西インド諸島サンサルバドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきっていた。
```

```
コロンブスは、その後もジパングをめざし、数回の航海でキューバ島やイスパニオラ島を発見した。しかし、最後までジパングを見つけることはできなかった。
```

```
~  
~  
:%s/...2年/1492年/g
```

ファイルの保存と vi の終了

ファイルの編集後、コマンドモードで（動作モード表示は消えます）、:（コロン）に続けて w (write) キーを入力します。入力した「:w」の文字列は画面最下行（ステータス行）に表示されます。ここで Return キーを入力すると、vi はファイル名、行数、バイト数を表示してファイルを保存します。

1492年、クリストファ・コロンブスは、スペイン女王の援助を受け、黄金に輝く地、ジパング（日本）を求めてスペインのパロスを出発した。3箇月後、西インド諸島サンサルバドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきっていた。

コロンブスは、その後もジパングをめざし、数回の航海でキューバ島やイスパニオラ島を発見した。しかし、最後までジパングを見つけることはできなかった。

~

~

“コロンブスのジパング” [新規ファイル] 10 行, 395 バイト

ファイルを変更していない状態またはファイルの保存後に vi を終了するときには、コマンドモードで:(コロン) キーのあとに q (quit) キーを入力します。ファイルの保存と同時に vi を終了するときには、wq を入力します。変更したファイルを保存せずに q キーを入力すると、次のような警告が出されます。

1492年、クリストファ・コロンブスは、スペイン女王の援助を受け、黄金に輝く地、ジパング（日本）を求めてスペインのパロスを出発した。3箇月後、西インド諸島サンサルバドルに到着した彼は、ここが東洋であり、南にはジパングがあると信じきっていた。

コロンブスは、その後もジパングをめざし、数回の航海でキューバ島やイスパニオラ島を発見した。しかし、最後までジパングを見つけることはできなかった。

~

~

変更後保存していません (:quit! で強制実行します)。¹

1. この行は反転表示されます。

このような場合、警告を無視して vi を終了するときには、q! または quit! と入力します。

オンラインマニュアルの参照

SunOS 上では、国際化された man コマンドと nroff およびその関連コマンドにより、日本語のオンラインマニュアルページを参照できます²。日本語のオンラインマニュアルページおよびそのためのマクロパッケージは、JFP によって提供されます。

マニュアルページの参照

環境変数 LC_MESSAGES が日本語ロケール (ja または japanese) に設定されているか、LC_MESSAGES が設定されておらず LANG が日本語ロケールに設定されているれば、man コマンドで指定されたマニュアルページが日本語で存在している限り、日本語のマニュアルページが表示されます。例として、dumpcs (1) のマニュアルページを参照してみましょう。次のように入力します。

```
sun% man dumpcs
マニュアルを清書中です。しばらくお待ちください...
```

しばらくすると dumpcs (1) の日本語マニュアルページが表示されます。

```
マニュアルを清書中です。しばらくお待ちください... 終了
dumpcs (1)          ユーザコマンド          dumpcs (1)

【名前】
  dumpcs - 現在のロケールのコードセット一覧表を表示

【形式】
  dumpcs [ -0123vw ]

【機能説明】
  dumpcs は、ユーザの現在のロケールに対応するコードセットの印
--継続-- (9%)
```

一時的に英文のマニュアルページを参照したい場合には、env (1) コマンドを使い、次のように「env LANG=C man dumpcs」と入力します。

2. ただし、troff の国際化は行われていないため、man コマンドの -t オプションを日本語マニュアルページの troff 出力に使うことはできません。

```
sun% env LANG=C man dumpcs
Reformatting page. Wait...
```

```
Reformatting page. Wait... done
dumpcs(1)          USER COMMANDS          dumpcs(1)
NAME
  dumpcs - show the codeset table for the current locale
SYNOPSIS
  dumpcs [ -0123vw ]
DESCRIPTION
  dumpcs shows a list of printable characters for the user's
  --More-- (9%)
```

和文と英文のマニュアルページを両方とも頻繁に参照する場合は、"env LANG=C man" の部分を C シェルの別名 (alias) として登録しておくといでしょう。

清書済みオンラインマニュアルと windex の作成

オンラインマニュアルは、システムの出荷時には nroff(1) または troff(1) のソース形式で提供されます。catman(1M) コマンドによって、これらのマニュアルページをあらかじめ清書しておく、man(1) コマンドによる表示の際の清書作業にかかる時間を節約できます (ただし、これによってさらにディスクスペースが消費されることに注意してください)。

また、catman コマンドは、whatis(1) および apropos(1) コマンドによって検索されるシステム機能の要約情報のデータベースファイル windex を作成します。-n オプションを用いるとこのデータベースは作成されず、-w オプションを用いるとこのデータベースの作成のみを行います。catman コマンドは、ユーザーが現在使用しているロケール名のディレクトリがオンラインマニュアルページのルートディレクトリに存在すると、その下のマニュアルページだけを清書します (たとえば、/usr/share/man/ja/man* がある場合は ja/man* を清書します)。英語のマニュアルページに対して常に翻訳マニュアルが用意されているとは限りませんので、必要に応じて次の例のように、ユーザーの通常の作業環境のロケールと、それが C ロケールではない場合には C ロケールと、それぞれでスーパーユーザーになって catman を実行してください。

```
sun# setenv LANG C
sun# catman 1 2 3
sun# setenv LANG ja
sun# catman 1 2 3
```

whatis による機能要約の参照

コマンドの機能の要約文だけを見たい場合には、whatis コマンドを使います。環境変数 LC_MESSAGES が日本語ロケール ja、ja_JP.PCK または ja_JP.UTF-8 に設定されているか、LC_MESSAGES が設定されておらず LANG が日本語ロケールに設定されており、whatis コマンドで指定されたマニュアルページが日本語で存在し、その情報を含む windex データベースが作られていれば、日本語のコマンド機能要約を次のように表示できます。

```
sun% whatis dumpcs
dumpcs      dumpcs (1)    - 現在のロケールのコードセット一覧表を表示
sun%
```

apropos によるキーワード検索

whatis とは逆に、コマンド機能要約文中のキーワードからコマンド名を探したいときには apropos コマンドを使用できます。環境変数 LC_MESSAGES が日本語ロケール ja、ja_JP.PCK または ja_JP.UTF-8 に設定されているか、LC_MESSAGES が設定されておらず LANG が日本語ロケールに設定されており、apropos コマンドで指定されたマニュアルページが日本語で存在し、その情報を含む windex データベースが作られていれば、日本語のコマンド機能要約を次のように表示できます。

```
sun% apropos コードセット一覧表
dumpcs      dumpcs (1)    - 現在のロケールのコードセット一覧表を表示
sun%
```

電子メールの送受信

JFP は、複数バイト言語国際化のために拡張された mailx (/usr/SUNWale/bin/mailx) を提供しています。以降、簡単な日本語メールの送受信を例にして、mailx の使い方とコード変換について説明します。

説明の中で、いくつかの mailx のコマンドと変数が登場しますが、それらの詳しい使い方については、国際化のために拡張された mailx(1) のマニュアルページまたは『OpenWindows ユーザーズガイド (上級編)』を参照してください。

メールの起動

まず、第3章の28ページの「コマンド検索パスの設定」を参考にし、`/usr/SUNWale/bin/mailx` を起動するように設定します。以下に述べるコード変換は、`LC_CTYPE` のロケールが環境変数 `LANG` または `LC_CTYPE` を介して `ja` または `japanese` に設定されている場合に行われます。また、環境変数 `LC_MESSAGES` が `ja` または `japanese` に設定されている場合は、`mailx` からの警告メッセージなどは日本語となります (メッセージ自体は JFP が提供します)。

メッセージの作成

メールを送るときは、`mailx` の引数に送り先のユーザー名を指定します。たとえば、`grape` というユーザーにメールを送る場合は、次のように指定します。

```
lemon% mailx grape
Subject:
```

`mailx` は `Subject:` の入力を要求してきます。`Subject:` にメールの用件を入力した後、メールのメッセージ本体を入力します。JFP の提供する日本語機能などを利用して日本語のメッセージを入力できます。

```
lemon% mailx grape
Subject: Next Lecture
葡萄さん：
次回の講習会は
    10月1日(木) 午後2:00開始
に変更されました。
```

メッセージをテキストエディタを使って作成することもできます。次のように `~v` を実行すると `vi` が起動されます。起動されるエディタは `VISUAL` 変数で指定します。デフォルトは `vi` です。`mailx` の変数は `mailx` を起動中、または起動ファイル (通常、ユーザーのホームディレクトリにある `.mailrc`) 内で設定できます。シェルの変数としても設定できますが、`mailx` を起動中の値および起動ファイルの値が優先されます。


```
Subject: Next Lecture
葡萄さん：
次回の講習会は
    10月1日（木） 午後2：00開始
に変更されました。
~v
```

~v を実行すると、ただちに vi の画面に変わります。

```
To: grape
Subject: Next Lecture

葡萄さん：
次回の講習会は
    10月1日（木） 午後2：00開始
に変更されました。
~
~
~
~
~
~
~/tmp/Re8570" 7行、121バイト
```

ここで、日本語を追加入力してみます。

```
To: grape
Subject: Next Lecture

葡萄さん：
次回の講習会は
    10月1日（木） 午後2：00開始
に変更されました。
会場は未定です。

檸檬
~
~
~
~
~
~
~
:wq
```

vi を終了すると元の mailx に戻ります。しかし、ここでは vi での編集イメージが画面に残っています。作成したメッセージを始めから見たい場合は、~p を実行するとメッセージが再表示されます。メッセージのサイズが大きい場合は画面が流れ

てしまい、内容をゆっくり読むことができません。それを防ぐために crt 変数と PAGER 変数をあらかじめ設定しておき、ページごとに表示させる方法があります。たとえば、mailx コマンド起動中に以下を設定します。

```
? set crt=20
? set PAGER=/usr/bin/more
```

行頭の?は mailx からのプロンプトです。

このような設定は、ユーザーのホームディレクトリにある .mailrc の中であらかじめ記述しておくとも mailx 起動時に常に実行されるので便利です。

```
~P
-----
メッセージの内容:
To: grape
Subject: Next Lecture

葡萄さん:
次回の講習会は
    10月1日(木) 午後2:00開始
に変更されました。
会場は未定です。

檸檬
(つづく)
```

作成したメールは、行の先頭での「.」(ピリオド)入力または CTRL-D により送信されます。mailx は終了してシェルに戻ります。メールの作成は日本語 EUC で行われてきましたが、送信時は7ビット JIS コードに変換されてから送られます。

```
~P
-----
メッセージの内容:
To: grape
Subject: Next Lecture

葡萄さん:
次回の講習会は
    10月1日(木) 午後2:00開始
に変更されました。
会場は未定です。

檸檬
(つづく)
.
作成終了
lemon%
```

メール作成の中止

作成の途中でメール送信を中止したいときは、割り込みキー (通常、CTRL-C) を入力してください。その場合、1 回目の割り込みキー入力で本当に途中で中止してよいかどうか尋ねてきます。そこでもう一度割り込みキーを入力すると、作成は中止されシェルに戻ります。このとき、途中まで作成されていたメッセージの内容が、`dead.letter` と呼ばれるファイル (あらかじめ設定されていない場合はユーザーのホームディレクトリにある `dead.letter`) に日本語 EUC で確保されます。

```
lemon% mailx grape
Subject: Next Lecture
葡萄さん：
  回の講習会は
    10月1日(木) 午後2:00開始
に変更されました。^C
(割り込み -- メッセージ送信をやめるにはもう一度)
^C"/home/garden/lemon/dead.letter" 4/63
lemon%
```

送信したメッセージの保存

送信に先立ち `record` 変数が設定されていると、送信するメッセージが `record` 変数で指定したフォルダに保存されます。`record` はデフォルトでは設定されていません。また、`record` 変数が設定されている場合に、`mailx` の変数 `folderconv` が設定されていると日本語 EUC で保存され、設定されていないと転送時の変換イメージ (7 ビット JIS コード) で保存されます。`folderconv` はデフォルトで設定されています。たとえば、`record` フォルダが `~/.record` の場合は以下のようにして、すでに送信したメッセージを参照できます。

```
lemon% mailx -f ~/.record
mailx version ALE5.0 Mon Sep 12 11:09:29 JST コマンドの概要は ? を
タイプしてください。
"+.record": 2 個のメッセージがあります。
>O 1 lemon Tue Sep 29 21:33 13/249 Excursion
O 2 lemon Tue Sep 29 21:47 15/245 Next Lecture
?
```

行の先頭の ? は `mailx` からのプロンプトです。現在 2 個のメッセージがあります。先程転送したメール (Subject: Next Lecture) は 2 番目にあります。メッセージを読むコマンドはいくつかありますが、ここではメッセージ番号をそのまま入力する方法で実行します。

```
? 2
Message 2:
From lemon Tue Sep 29 21:47:01 1992
To: grape
Subject: Next Lecture
```

葡萄さん：
次回の講習会は
10月1日（木） 午後2：00開始
に変更されました。
会場は未定です。

檸檬

?

lemon%

メッセージの読み取り

今度は、先程送ったメールをユーザー `grape` 側から読んでみます。メールボックスに届いたメールを読むには、`mailx` を引数なしで起動します。メールボックスにメールが届いていると、次のように表示されます。なお、この場合も `LC_CTYPE` および `LC_MESSAGES` が環境変数 `LANG`、または `LC_CTYPE` および `LC_MESSAGES` を介して `ja` または `japanese` に設定されているものとします。

```
grape% mailx
mailx version ALE5.0 Mon Sep 12 11:09:29 JST コマンドの概要は ? をタイプ
してください。
"/var/mail/grape": 2 個のメッセージがあります。 2 個新規。
> N 1 Y.Suzuki (Tokyo off Tue Sep 29 21:40 17/528 President's Report
  N 2 A.Lemon Tue Sep 29 21:47 21/513 Next Lecture
?
```

メッセージは日本語 EUC ではなく、7 ビット JIS コードに変換されて到着していますが、表示される時には自動的に日本語 EUC にコード変換されます。

```
? 2
Message 2:
From lemon Tue Sep 29 21:47 JST 1992
Date: Tue, 29 Sep 92 21:47:02 JST
From: lemon (A.Lemon)
To: grape
Subject: Next Lecture
```

```
葡萄さん：
次回の講習会は
    10月1日（木） 午後2：00開始
に変更されました。
会場は未定です。
```

```
檸檬
```

```
?
```

この場合も72ページの「メッセージの作成」の項で行なったように、変数 `crt` および `PAGER` を設定しておく、大きなサイズのメッセージも画面上で流れることなく、ページごとに見ることができます。

到着したメッセージの保存

到着したメッセージを保存するために、いくつかの `mailx` のコマンドが用意されています。 `s` (`save`)、 `c` (`copy`)、 `w` (`write`) コマンドは、変数 `folderconv` が設定されていると到着したメッセージを日本語 EUC にコード変換して保存します。 `save` と `copy` はフォルダの形式でヘッダ情報をつけて保存します。 `write` はメッセージ本体だけを保存します。また、 `save` を実行するとそのメッセージはメールボックスから取り除かれますが、 `copy` の場合は保存後もメールボックスに残ります。

```
? save 2 folder.1
"folder.1" [新規ファイル] 22/481
```

`save` コマンドにより、メッセージ番号 2 のメッセージは `folder.1` に保存されました。このときは日本語 EUC にコード変換されています。到着したメッセージを変換しないでそのままフォルダに保存したいときは、 `nsave` または `ncopy` コマンドを利用して保存します。

```
? nsave 2 folder.2
"folder.2" [新規ファイル] 22/523
```

nsave コマンドにより、メッセージ番号 2 のメッセージは folder.2 にコード変換されずに保存されました。

一度読んだメールは特に明示的に保存しないと、mailx 終了時に mbox と呼ばれるフォルダ (あらかじめ設定していない場合はユーザーのホームディレクトリにある mbox) に保存されるように設定されています (この設定は mailx の変数の値と終了コマンドの種類によって変わります)。mbox に保存される場合も、日本語 EUC にコード変換されてから保存されます。

到着したメッセージの編集

```
? visual 2
```

v (visual) コマンドにより、到着したメッセージをテキストエディタを使って編集できます。起動されるエディタは VISUAL 変数で指定されます。デフォルトは vi です。このとき、編集するメッセージは日本語 EUC にコード変換された上でエディタが起動されます。

```
From lemon Tue Sep 29 21:47 JST 1992
Return-Path: <lemon> by sushi.Japan.Sun.COM (5.0/SMI-SVR4)
 id AA01421; Tue, 29 Sep 92 21:47:02 JST
Date: Tue, 29 Sep 92 21:47:02 JST
From: lemon (A.Lemon)
Message-Id: <9209300939.AA01421@sushi.Japan.Sun.COM>
To: grape
Content-Type: text
Content-Length: 112
Subject: Next Lecture
X-Sun-Charset: ja_JP.euc
```

```
葡萄さん：
  次回の講習会は
    10月1日(木) 午後2:00開始
に変更されました。
会場は未定です。
```

```
檸檬
~
~
~
"/tmp/Rz8575" 18 行, 412 バイト
```

次のように、実際にメッセージを変更すると、変更後は日本語 EUC でメールボックスに保存されます。

```
From lemon Tue Sep 29 21:47 JST 1992
Return-Path: <lemon> by sushi.Japan.Sun.COM (5.0/SMI-SVR4)
id AA01421; Tue, 29 Sep 92 21:47:02 JST
Date: Tue, 29 Sep 92 21:47:02 JST
From: lemon (A.Lemon)
Message-Id: <9209300939.AA01421@sushi.Japan.Sun.COM>
To: grape
Content-Type: text
Content-Length: 112
Subject: Next Lecture
X-Sun-Charset: ja_JP.euc
```

**** 予定変更のお知らせ ****

葡萄さん：
次回の講習会は
10月1日（木） 午後2：00開始
に変更されました。
会場は未定です。

檸檬
~
~
~
~
:wq

メッセージの返答方法

76ページの「メッセージの読み取り」に戻り、返事を出してみます。

```
? 2
Message 2:
From lemon Tue Sep 29 21:47 JST 1992
Date: Tue, 29 Sep 92 21:47:02 JST
From: lemon (A.Lemon)
To: grape
Subject: Next Lecture
```

葡萄さん：
次回の講習会は
10月1日（木） 午後2：00開始
に変更されました。
会場は未定です。

檸檬

```
? r
To: lemon
Subject: Re: Next Lecture
```

r コマンドを実行すると、自動的に差出人宛てのメールを作成する状態になります。ここからは72ページの「メッセージの作成」の項で示したとおりに、~p、~v などを利用してメッセージを作成できます。返事の中に、現在読んでいるメッセージまたは別のメッセージを挿入したいときは、~m または ~f を実行します。どちらの場合も、挿入メッセージが日本語 EUC にコード変換されて挿入されます。

```
? r
To: lemon
Subject: Re: Next Meeting

~m
メッセージ挿入: 2
(つづく)
```

~m の入力時に特にメッセージ番号を指定しなかったので、現在読んでいる 2 番のメッセージが挿入されました。つづいて vi を起動して (~v コマンドの実行) 返事を書きます。

```
To: lemon
Subject: Re: Next Lecture

    From lemon Tue Sep 29 21:47 JST 1992
    Date: Tue, 29 Sep 92 21:47:02 JST
    From: lemon (A.Lemon)
    To: grape
    Subject: Next Lecture

    葡萄さん：
    次回の講習会は
    10月1日（木） 午後2：00開始
    に変更されました。
    会場は未定です。

    檸檬

了解しました。

葡萄
~
~
~
:wmq
```

挿入したメッセージは、各行の先頭にタブが挿入され段落がつけられています。この自動的に挿入される文字列は、変数 indentfix で設定されます。デフォルトはタブです。また、この文字列の段落付けは ~f でメッセージを挿入するときには適用されません。


```
:wq
"/tmp/Re8580" 21 行、333 バイト
(つづく)
.
作成終了
?
```

72ページの「メッセージの作成」の項で示したとおりに作成を終了すると、メッセージは7ビット JIS コードに変換され送信されます。mailx 自身の終了には、quit、コマンド行の先頭での CTRL-D、exit などのコマンドが用意されています。quit と CTRL-D は、既読のメッセージを mbox に保存して未読のメッセージをメールボックスに保存します。exit はメールボックスを変更せず、mbox に保存することなく mailx を終了します。mailx 終了の動作は、hold などの変数を設定してカスタマイズできます。

メール到着の通知

ここでは、comsat を利用したメール到着の通知を紹介します。まず、到着した日本語メールが正しくコード変換して表示されるように、第4章「日本語対応 SunOS のシステム環境設定」の36ページの「in.comsat の設定」に従って inetd(1M) の構成を修正してください。

次に、biff(1B) でメール受信の通知を行うように設定します。

```
grape% biff y
```

このように設定すると、7ビット JIS コードに変換されている日本語メールが到着するときに、日本語 EUC にコード変換されて画面にはじめの数行が表示されます。

```
New mail for grape has arrived:
----
Date: Wed, 30 Sep 92 11:56:35 GMT
From: lemon (A. Lemon - Garden Company)
To: grape
Subject: Next Lecture
```

```
葡萄さん：
次回の講習会は
...以下省略...
```


日本語環境用 SunOS コマンド

この章では、JFP によって日本語環境用に提供される SunOS コマンドについて説明します。

フィルタを用いたコード変換

現在、日本語文字コードには JFP で採用している日本語 EUC、PCK のほかに、JIS コード、UTF8 などがあります。

Sun のネットワーク機能を使って、他のマシンからファイルをコピーしたり、また他のマシンへファイルを転送したりする場合、採用している日本語文字コードが違ふとそのままでは文字化けを起こして使用できません。日本語 EUC から JIS コードへ、またはその逆など相互にファイル中の漢字コードを変換するフィルタが、このコード変換フィルタです。

Solaris では、『X/Open Portability Guide Issue 4.2 (XPG4.2)』規定に準拠した `iconv` コマンドがサポートされています。`iconv` は次のような書式で使われます。

```
sun% iconv -f <変換元文字コード> -t <変換先文字コード> [filename...]
```

JFP では、この `iconv` に対応した、日本語文字コード用の変換モジュールを提供しています。`iconv` に以下のように指定して実行すると、変換が実現されます。`iconv` コマンドについては、`iconv(1)` および `iconv_ja(5)` のマニュアルページも参照してください。また、`iconv(3)` インタフェースを使用したい開発者

の方は、『JFP 開発ガイド』の第 4 章「日本語文字コード変換」もあわせて参照してください。

表 6-1 iconv コマンドによる変換

変換元文字コード	変換先文字コード
eucJP	PCK
eucJP	ISO-2022-JP
eucJP	ISO-2022-JP.RFC1468
eucJP	JIS7
eucJP	SJIS
eucJP	UTF-8
eucJP	UTF-8-Java
eucJP	jis
eucJP	ibmj
SJIS	eucJP
SJIS	ISO-2022-JP
SJIS	UTF-8
SJIS	jis
SJIS	ibmj
PCK	eucJP
PCK	UTF-8
PCK	UTF-8-Java

表 6-1 iconv コマンドによる変換 続く

PCK	ISO-2022-JP
PCK	ISO-2022-JP.RFC1468
PCK	jis
PCK	ibmj
ISO-2022-JP	eucJP
ISO-2022-JP	PCK
ISO-2022-JP	SJIS
ISO-2022-JP	UTF-8
UTF-8	eucJP
UTF-8	SJIS
UTF-8	PCK
UTF-8	ISO-2022-JP
UTF-8	ISO-2022-JP.RFC1468
UTF-8-Java	eucJP
UTF-8-Java	PCK
JIS7	eucJP
jis	eucJP
jis	PCK
jis	SJIS
ibmj	eucJP

表 6-1 iconv コマンドによる変換 続く

ibmj	PCK
ibmj	SJIS

上記の文字コードの意味は以下のとおりです。

文字コード	意味
eucJP	日本語 EUC
PCK	PCK
SJIS	PCK と同じ
ISO-2022-JP	ISO 2022 情報交換用符号による G0 への指示機能を用いて "UI-OSF 日本語環境実装規約 Version 1.1" の規定に従い表現された、ISO 646 IRV または JIS X 0201、JIS X 0208、および JIS X 0212
ISO-2022-JP.RFC1468	ISO 2022 情報交換用符号による G0 への指示機能を用いて RFC1468 (Request for Comments: 1468 Japanese Character Encoding for Internet Messages) の規定に従い表現された、ISO 646 IRV または JIS X 0201-1976 (片仮名用図形 (キャラクタ集合を除く) および JIS X 0208-1983
JIS7	ISO-2022-JP と同じ
jis	JLE、JFP 2.4 以前に提供されていた 7 ビット JIS コード
ibmj	IBM 漢字コード
UTF-8	UNICODE
UTF-8-Java	Java で実装されている UNICODE

JLE 1.x からの互換性を持つコード変換フィルタのコマンドとして、次のものがあります。

表 6-2 コード変換フィルタコマンド

コマンド	機能
<code>jistoenc</code>	JIS コードを日本語 EUC に変換
<code>jistosj</code>	JIS コードを PCK に変換
<code>euctojis</code>	日本語 EUC を JIS コードに変換
<code>euctosj</code>	日本語 EUC を PCK に変換
<code>sjtojis</code>	PCK を JIS コードに変換
<code>sjtoenc</code>	PCK を日本語 EUC に変換
<code>ibmjtoenc</code>	IBM 漢字コードを日本語 EUC に変換
<code>euctoibmj</code>	日本語 EUC を IBM 日本語コードに変換

ただし、これらには以下の制限事項があるため、通常は `iconv(1)` コマンドでの使用をお勧めします。

- 日本語 EUC との変換を行うコマンドでは、JIS X 0212-1990 (いわゆる補助漢字) をサポートしていません。
- PCK との変換を行うコマンドでは、『TOG (The Open Group) 日本ベンダ協議会推奨 日本語 EUC・シフト JIS 間コード変換仕様』に基づく変換をサポートしていません。『TOG 日本ベンダ協議会推奨 日本語 EUC・シフト JIS 間コード変換仕様』に関しては、付録 A を参照してください。
- JIS との変換を行うコマンドでは、更新番号を含むエスケープシーケンスに対応していません。

漢字端末と日本語入力

TTY 環境に構成に使用するコマンドは `stty`、`setterm`、および `jtty` です。これらのコマンドの使用については、各マニュアルページを参照してください。ここで

は日本語端末を `seeterm` で構築する上で使用される、日本語文字コード変換 STREAMS モジュールについて説明します。

JFP は、端末ベースの入出力に対して日本語の処理機能を提供しています。その中でいろいろな種の端末とのインタフェースを取るために `setterm` で端末を設定することによって扱う文字コード (ロケールにより設定される) と端末の文字コードの違いをコード変換することによって吸収したり、STREAMS 内部で日本語文字コードが正しく扱うことができるような設定をしたりします。

たとえば、`ja` ロケールで PCK 漢字端末が接続されている場合、入力 は日本語 EUC に変換され、端末への出力は PCK に変換されます。

漢字端末を設定する `setterm` コマンドは、次のような書式で使用します。

```
sun% setterm [ -x name ] [ -t ]
```

日本語文字コード変換 STREAMS モジュール

JFP では、`setterm` コマンドで漢字端末を設定できるように、以下のような日本語文字コード変換 STREAMS モジュールを提供しています。

表 6-3 日本語文字コード変換 STREAMS モジュール

コード変換 STREAMS モジュール	説明
<code>jconvS</code>	PCK 端末用
<code>jconv7</code>	7 ビット JIS 端末用
<code>jconv8</code>	8 ビット JIS 端末用
<code>jconvu</code>	UTF-8 端末用
<code>jconvrs</code>	<code>ja_JP.PCK</code> ロケール用
<code>jconvru</code>	<code>ja_JP.UTF-8</code> ロケール用

現在のロケールと設定したい端末名に応じて、適当な STREAMS モジュールが `setterm` コマンドを通して設定されます。以下は、`setterm` の `-x` オプションで指定可能な端末名です。

注・日本語ロケール間でロケールを切り替える場合は、その都度、端末設定も行うようにしてください。これらのロケール間では `push` される STREAMS モジュールが異なるため、再度 `setterm` が実行されないと正しく表示できなくなります。

表 6-4 `setterm` 端末名

端末名	説明
JapanPCK	PCK 端末用
PCK	
JapanSJIS	
SJIS	
JapanNewJIS8	8 ビット新 JIS 端末用
NewJIS8	
JapanNewJIS7	7 ビット新 JIS 端末用
NewJIS7	
JapanOldJIS8	8 ビット旧 JIS 端末用
OldJIS8	
JapanOldJIS7	7 ビット旧 JIS 端末
OldJIS7	
JapanEUC	日本語 EUC 端末用
GenericEUC	
EUC	
UTF-8	UTF-8 端末用

表 6-4 setterm 端末名 続く

ASCII	ASCII 端末用
Error	setterm を行なった際のエラー復旧用
sane	STREAMS のリセット用

mailx/in.comsat でのコード変換

複数バイト言語国際化のために拡張された mailx を用いると、日本語 EUC で書かれたメールメッセージは自動的に JIS コードに変換されて送信されます。このようにして変換されたメッセージは受信後、type、print などメッセージを画面に表示するコマンドにより EUC に変換されて表示されます。また、save、copy、write などのコマンドにより EUC に変換されて格納できます。この変換は、フォルダ内のコードを決定するために用意されたメール変数 folderconv が設定されている場合に実行されます。folderconv が設定されていない場合、メッセージはコード変換されずに格納されます。folderconv はデフォルトで設定されています。また、folderconv がどのような指定の場合でもコード変換をしないで格納するためのコマンドとして、nsave と ncopy が用意されています。

複数バイト言語国際化のために拡張された in.comsat は、到着メールを JIS コードから EUC に変換して表示します。

使用方法は、第 5 章「国際化 SunOS コマンドの利用の71ページの「電子メールの送受信」を参照してください。

パートII JFP - GUI



GUI 環境の起動

Solaris では、GUI 環境として日本語 OpenWindows と、Solaris 共通デスクトップ環境 (Common Desktop Environment) (以降 Solaris CDE とします) を用意しています。

ログインサーバー

「エンドユーザシステムサポート」のクラスタ以上をインストールした場合、システムの起動後に図 7-1 のようなログイン画面が表示され、ユーザーはウィンドウ環境 (Solaris CDE または OpenWindows) やログイン時のロケールなどを簡単に選択できます。

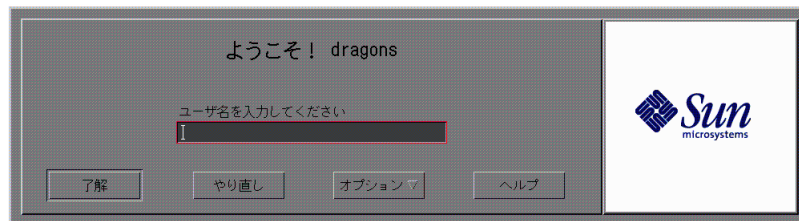


図 7-1 ログイン画面

ログインサーバー (dtlogin) が実行されるロケールは、デフォルトの状態では、インストール時に設定したシステムのデフォルトロケールです。

セッション

「オプション」メニューの「セッション」でウィンドウ環境を選択できます。デフォルト値は、「ユーザが最後に使ったデスクトップ」です。最初のログインでは、次のようなウィンドウが表示されます。

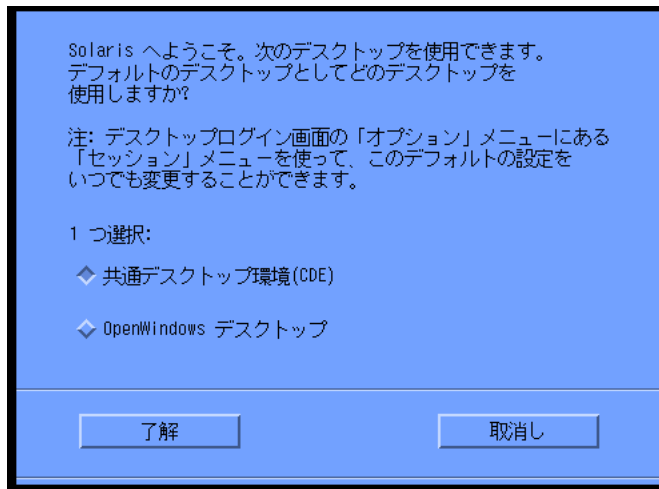


図 7-2 最初のログイン画面

言語

「オプション」メニューの「言語」で、ユーザーがログインする際に使用するロケールを選択できます。日本語環境では、ja、japanese、ja_JP.PCK および ja_JP.UTF-8 の 4 種類のロケールから選択できます。なお、japanese は JLE 1.1.3 以前のシステムとの互換性のためのロケールで ja と同様に日本語 EUC を扱います。日本語 EUC を使用する場合は、ja ロケールを使うようにしてください。



図 7-3 ロケールの選択画面

Solaris CDE と日本語 OpenWindows 環境 に共通の機能

日本語入力システムの切り替え方法

日本語入力システムは、Wnn6、ATOK8、cs00 の 3 種類から選択できます。Solaris CDE および OpenWindows のどちらも、ワークスペースメニューから日本語入力システムを切り替えることができます。なお、新しい日本語入力システムを有効にするには、一度 GUI セッションからログアウトして、再度ログインし直す必要があります。

次に、Solaris CDE と OpenWindows のワークスペースメニューを示します。

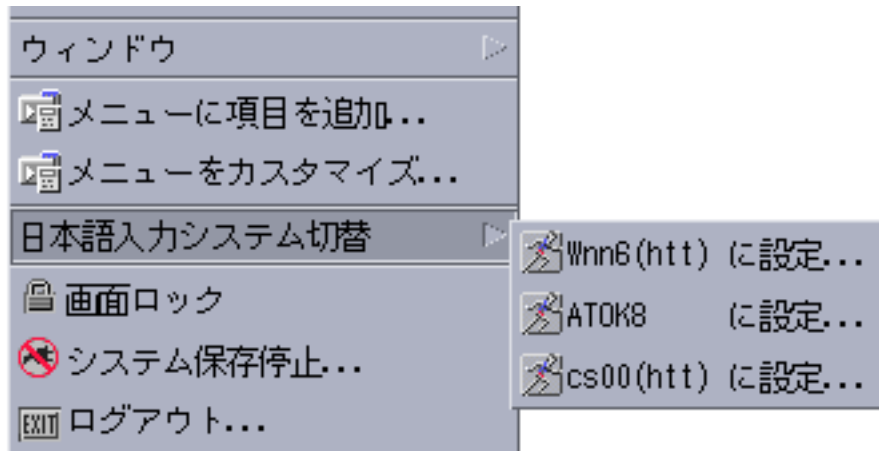


図 8-1 Solaris CDE ワークスペースメニュー

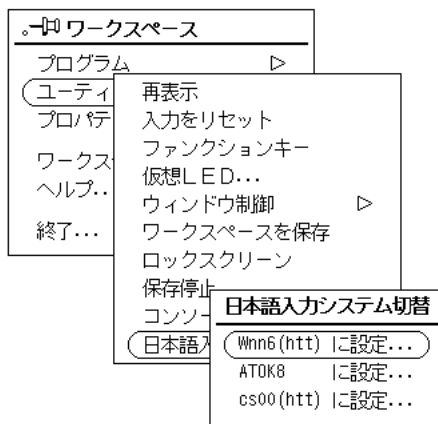


図 8-2 日本語 OpenWindows ワークスペースメニュー

日本語入力サーバー(htt)

Solaris の日本語入力サーバー htt は、X11R5 XIMP4.0 プロトコルに加えて、X11R6 XIM プロトコルをサポートしています。詳細は、htt(1) のマニュアルページを参照してください。

日本語フォント

Solaris では、ウィンドウ環境で使用される日本語フォントに関して次の機能が提供されます。

- 日本語 TrueType フォントのサポート
- 日本語ビットマップフォント
- JIS X 0208-1990 文字集合のサポート
- JIS X 0212-1990 補助漢字のサポート
- IBM 拡張文字と NEC 選定ユーザー定義文字のサポート
- ユーザー定義文字フォントのサポート
- XGL™ 漢字ストロークフォント

日本語フォントに関する技術的な情報については、『JFP 開発ガイド』を参照してください。

日本語 TrueType フォント

Solaris では、X Window System と Adobe Display PostScript™ 上で日本語 TrueType フォントを使用できます。これに伴い、株式会社リコーが開発した HG ゴシック体 B、HG 明朝体 L、および平成明朝体 W3H の TrueType フォントを新たに提供します。これらのフォントは、Solaris CDE および OpenWindows 環境上でアウトラインフォントとして使用できます。

- HG ゴシック体 B
XLFD 名 : -ricoh-hg gothic b-medium-r-normal- -*
DPS でのフォント名 : HG-GothicB
- HG 明朝体 L
XLFD 名 : -ricoh-hg mincho l-medium-r-normal- -*
DPS でのフォント名 : HG-MinchoL
- 平成明朝体 W3H
XLFD 名 : -ricoh-heiseimin-w3-r-normal-*
DPS でのフォント名 : HeiseiMin-W3H

また、HG ゴシック体 B と HG 明朝体 L は、それぞれ中ゴシック BBB (GothicBBB-Medium) とリュウミン L-KL (Ryumin-Light) として、X Window System と Adobe Display PostScript 上で利用できます。

日本語ビットマップフォント

次の日本語ビットマップフォントが提供されています。

表 8-1 JFP で提供される日本語ビットマップフォント

字体	サイズ (単位はポイント)
gotm (ゴシック体)	12、14、16、20、24
gotb (ゴシック体ボールド)	12、14、16、20、24
minm (明朝体)	16、20、24
minb (明朝体ボールド)	16、20、24
hg gothic b (HG ゴシック体 B)	10、12、14、16、18、20、24
hg mincho l (HG 明朝体 L)	16、18、20、24
heiseimin (平成明朝体 W3H)	10、12、14、16、18、20、24

これらのフォントに X プログラムからアクセスする際に使用する XLFD 名については、`xlsfonts` コマンドで調べることができます。

Solaris CDE 環境では、スタイルマネージャを使って、使用するフォントを簡単に切り替えることができます。詳細は、第 9 章「Solaris CDE の機能概要」を参照してください。

JIS X 0208-1990 文字集合

JIS X 0208-1990 年度版文字集合のサポートとして、85 区 5 点 (JIS 0x7425) および 6 点 (JIS 0x7426) の 2 文字を使用できます。なお、X プログラムからアクセスする際に使用する XLFD 名のレジストリフィールドは、既存のアプリケーションとの互換性のため、`jisx0208.1983` になっています。

JIS X 0212-1990 補助漢字

JIS X 0212-1990 補助漢字用のフォントとして、株式会社リコーの開発した平成明朝体 W3 を提供しています。このフォントは、Solaris CDE および OpenWindows 環境上で、X Window System と Adobe Display PostScript アプリケーションから使用できます。

IBM 拡張文字と NEC 特殊文字

マイクロソフト標準キャラクタセットで使用されている IBM 拡張文字 (SJIS 0xfa40 - 0xfc4b) と NEC 特殊文字 (13区、SJIS 0x2d20 以降) を Solaris でも使用できます。

ユーザー定義文字フォント

Solaris では、ユーザー定義文字は既存のフォントファイルとは別のユーザー定義文字フォントファイルとして取り扱います。ユーザー定義文字を登録するには「Solaris 外字ツール (sdtudctool)」を使用します。「Solaris 外字ツール」の詳細は、第 14 章を参照してください。

XGL 漢字ストロークフォント

XGL 実行環境で使用できる漢字ストロークフォント `Kanji.font` が提供されています。フォントは `/opt` ディレクトリにインストールされます。XGL とストロークフォントの使用の詳細は、『XGL Programmer's Guide』を参照してください。

フォントのディレクトリ構造

日本語表示用のフォントは、`/usr/openwin/lib/locale/ja/X11/fonts` ディレクトリに存在します。表 8-2 に、このディレクトリの構成について示します。

表 8-2 /usr/openwin/lib/locale/ja/X11/fonts/ の構成

ディレクトリ	ファイル拡張子	内容
75dpi	*.pcf	ビットマップフォント
TT	*.ttf	TrueType フォント
TTbitmap	*.pcf	TrueType ビットマップフォント

EUC コードセット

日本語 Solaris CDE (以降、日本語 SCDE とします) と日本語 OpenWindows では、日本語ロケールの文字集合として、コードセット 0、1、2 に加えて、コードセット 3 (JIS X 0212-1990 補助漢字) をサポートします。

なお、印刷方法については、第 4 章の 51 ページの「JIS X 0212、ユーザー定義文字、ベンダー定義文字の印刷サポート」を参照してください。

ja_JP.UTF-8 ロケール

日本語 SCDE と CDE/Motif アプリケーションでは、ja、ja_JP.PCK および ja_JP.UTF-8 の 3 つの日本語ロケールをサポートします。

注 - 日本語 OpenWindows および XView™/OLIT ライブラリを使用して書かれたアプリケーションは、ja_JP.PCK、ja_JP.UTF-8 ロケールでは動作しません。

ja と ja_JP.PCK ロケール間の文字変換

ja と ja_JP.PCK ロケールで動作する別のアプリケーション間で、文字列のカット & ペーストなどを実行した場合、付録 A「文字コード変換規則」に基づき、自動的にコードが変換されます。

日本語化されているプログラム

インストールツール

インストール用ユーザーインターフェースのラベルとヘルプメッセージが日本語で表示されます。使い方については、インストール関連のマニュアルを参照してください。

システム管理ツール

システム管理ユーザーインターフェース (admintool) のラベルとヘルプメッセージが日本語で表示されます。

グラフィックスライブラリ

XIL™ イメージ処理ライブラリと XGL グラフィックスライブラリのメッセージが日本語で表示されます。

Kodak Color Management System (KCMS)

Kodak Color Management System™ のライブラリと KCMS™ 調整ツール (kcms_calibrate) グラフィカルユーザーインターフェースのメッセージとヘルプが日本語で表示されます。

AccessX

AccessX は、X11 サーバーを拡張するためのグラフィカルユーザーインターフェースです。X11 が標準でサポートしているキーボードとマウスを使用するのが困難なユーザーを支援するための機能を提供します。AccessX では、ラベル、ヘルプメッセージ、およびマニュアルページが日本語で表示されます。

Motif

Motif ライブラリのメッセージが日本語で表示されます。

電源管理

電源管理 (dtpower) のラベルとヘルプメッセージが日本語で表示されます。

xserverconfig

xserverconfig のラベルが日本語で表示されます。

ToolTalk

ToolTalk™ ライブラリのメッセージとヘルプが日本語で参照できます。

仮想 LED

仮想 LED (vled) は、キーボードに付いている LED をソフトウェア的にシミュレートするユーティリティです。

日本語キーボードにはローマ字とかなを切り替えるロックキーが付いていますが、実際の LED は付いていません。仮想 LED を使って、このロックキーの状態を画面上で確認できます。

詳細は、第 13 章と vled(1) のマニュアルページを参照してください。

Solaris CDE の機能概要

Solaris CDE では、次の機能が提供されています。

- 日本語化されているアプリケーション
- 日本語環境固有のアプリケーション
- 日本語オンラインヘルプ
- 日本語オンラインマニュアル
- 日本語入力サーバー

日本語化されているアプリケーション

下記に示すアプリケーションでは、日本語の入力や日本語ファイルの表示など日本語を扱うことができます。また、ユーザーインタフェースのラベルやヘルプメッセージなどが日本語で表示されます。

注 - 「時計」は OpenWindows のツールのため、ja_JP.PCK、または ja_JP.UTF-8 ロケールで起動した場合、ラベルやメッセージは英語表記になります。

- ファイルマネージャ
- テキストエディタ
- メールプログラム
- カレンダー
- Web ブラウザ

- 端末エミュレータ
- コンソール
- 時計
- 電卓
- パフォーマンスメーター
- 電源管理ツール
- 印刷マネージャ
- オーディオ
- イメージビューア
- スナップショット
- アイコンエディタ
- アクション作成
- アプリケーションビルダ
- スタイルマネージャ
- アプリケーションマネージャ
- ヘルプマネージャ
- マニュアルページビューア
- テキストノート
- 音声ノート
- アドレスマネージャ
- ファイルを検索
- プロセスマネージャ

日本語環境固有のアプリケーション

Solaris CDE に固有のアプリケーションとして、次のプログラムが提供されています。

- Solaris 外字ツール

Solaris 外字ツール (sdtudctool) は Solaris CDE 上で、ユーザー定義文字を別フォントファイルとして登録するツールです。このツールを利用して、画面上で使用される複数のサイズのビットマップフォントにユーザー定義文字を一括して登録できます。また、同時にアウトラインフォントにもユーザー定義文字を登録できます。詳細は、第 14 章と第 15 章を参照してください。

■ cs00 ユーザー辞書ツール (sdtudicm)

Solaris CDE がサポートしている日本語入力システムの 1 つである cs00 のユーザー辞書を編集するユーティリティツールです。大きく分けて次の 2 つの機能を備えています。詳細は、『cs00 ユーザーズガイド』を参照してください。

- 利用中の辞書に対して単語の登録・削除を行う
- 利用中でないユーザー辞書形式ファイルを編集する

日本語オンラインヘルプ

次のいずれかの操作により、日本語ヘルプを参照できます。

- ◆ F1 キーまたは Help キーを押します。
- ◆ 各アプリケーションの「ヘルプ」メニューまたはヘルプボタンを選択します。
- ◆ フロントパネルからヘルプマネージャを起動します。
- ◆ フロントパネルのヘルプサブパネルまたは各アプリケーションの「ヘルプ」メニューから「アイテムヘルプ」を選択します。

詳細は、『Solaris 共通デスクトップ環境 ユーザーズガイド』を参照してください。

日本語 AnswerBook

ワークスペースメニューの「プログラム」またはフロントパネルの「ヘルプ」サブパネルから「AnswerBook2」を選択すると、AnswerBook2™ のビューアが起動し、日本語 AnswerBook™ を参照できます。

日本語入力システム

Solaris CDE では、日本語入力システムとして Wnn6、ATOK8、または cs00 を使用できます。ワークスペースメニューの「日本語入力システム切替」で変更できます。

デスクトップ環境のカスタマイズ

スタイルマネージャを起動して、次の項目をカスタマイズできます。

- ワークスペースのカラーとパレット
- フォントのサイズとフォントグループ
- ワークスペースの背景パターン
- キーボード (キーのリピート機能とキークリックボリューム)
- マウス
- ビープ音
- 画面ロックとスクリーンセーバー
- ウィンドウ
- セッションの開始および終了方法

フォントグループの設定方法

スタイルマネージャを使って、ユーザーが追加したフォントグループをデスクトップアプリケーションで使用できます。

たとえば、HG-Gothic-B を使用するには次の操作を実行します。

1. スタイルマネージャを起動し、「フォント」メニューを選択します。
「フォント」ダイアログボックスが表示されます。
2. 「フォントグループの追加」ボタンを押します。
図 9-1 のようなウィンドウが表示されます。

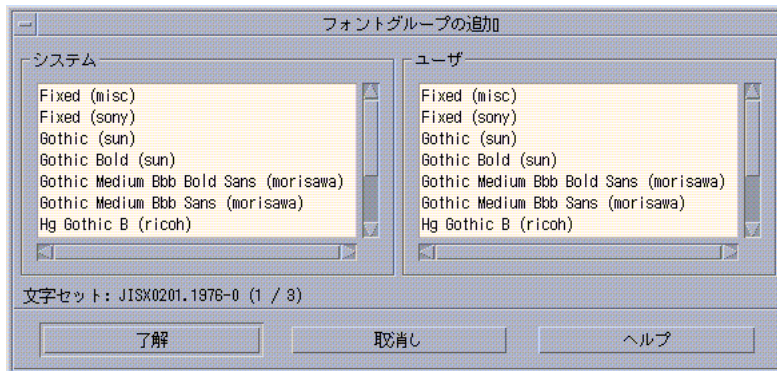


図 9-1 「フォントグループの追加」ウインドウ

3. 3種類の文字集合 (**JIS X 0201**、**JIS X 0208**、**JIS X 0212**) について、システムユーザーフォントとして、リストの中から「**Hg Gothic B**」を選択し、「了解」を押します。
4. フォントグループ名を入力します。
5. フォントグループのリストから109ページの手順 4 で入力したグループ名を選択します。
6. 「了解」を押し、アプリケーションを起動します。

プリンタの設定方法

ja_JP.UTF-8 ロケール環境で EUC 対応のプリンタにメールを出力するには、次の操作を実行します。

- ◆ \$HOME/.dt/types に dtmail.dt という名前で、次の内容のファイルを作成します。

```

ACTION Print
{
    LABEL          Print
    ARG_TYPE       DTMAIL_FILE
    TYPE           COMMAND
    WINDOW_TYPE    NO_STDIO
    EXEC_STRING     sh -c `¥
                    dtmailpr -p -f %(File)Arg_1% | iconv -f UTF-8 -t encJP | dtlp -u \
%(File) Arg_1%;`
}

```

Solaris CDE を使用する際の推奨環境

Solaris CDE では、ja、ja_JP.PCK および ja_JP.UTF-8 の 3 つのロケールをサポートしていますが、いずれか 1 つのロケール環境で使用することをお勧めします。1 人のユーザーが ja、ja_JP.PCK および ja_JP.UTF-8 ロケールを混在して使用すると、次の問題が起こります。

- 最初に CDE にログインしたときのロケールにあわせてワークスペースメニューが作成されるために、他のロケールでは、ワークスペースメニューが正しく表示されません。
- \$HOME/.dt/types に日本語ラベルのついたアクション、または日本語名のアクションファイルが存在する場合、そのアクションは、ja、ja_JP.PCK または ja_JP.UTF-8 ロケールのいずれかの環境でしか実行できません。
- ja ロケールで登録したカレンダーは、ja_JP.PCK または ja_JP.UTF-8 ロケールで起動されたカレンダー・マネージャで見ることができません。ja_JP.PCK または ja_JP.UTF-8 ロケールで作成した場合も同様に、他のロケールでは表示できません。

日本語 OpenWindows の機能概要

日本語 OpenWindows では、次のような機能が提供されています。

- 日本語化されているプログラム
- 日本語環境用プログラム
- ワークスペースプロパティ機能による日本語環境と英語環境の切り替え
- 日本語オンラインヘルプ
- 日本語オンラインマニュアル
- 日本語フォント
- 日本語対応ツールキットによる日本語プログラムの開発環境

日本語化されているプログラム

日本語 DeskSet プログラム

DeskSet™ 上で、日本語の入力や日本語ファイルの表示などの日本語を取り扱うことができます。DeskSet のボタン、メニュー、およびメッセージが必要に応じて日本語で表示されます。

また、DeskSet から日本語ファイルを日本語 PostScript 対応のプリンタに出力できます。

日本語対応された DeskSet は次のとおりです。

- コマンドツール

- オーディオツール
- テキストエディタ
- テープツール
- ファイルマネージャ
- バインダ
- メールツール
- スナップショット
- カレンダマネージャ
- アイコンエディタ
- 時計ツール
- 計算ツール
- イメージツール
- シェルツール
- 印刷ツール
- ヘルプビューワ
- パフォーマンスメーター

OPENLOOK ウィンドウマネージャ

OPENLOOK ウィンドウマネージャ (OLWM) のメニューとメッセージが日本語化されています。

ワークスペースプロパティとワークステーション情報

ワークスペースメニューから選択できるワークスペースプロパティ (props) とワークステーション情報 (wsinfo) のメッセージとヘルプが日本語で表示されます。

ツールキットのライブラリ

XView ツールキット、OLIT ツールキット、Motif ツールキットのライブラリのメッセージとヘルプが日本語で参照できます。

Motif ツールキットの日本語メッセージを利用する場合には、環境変数 NLSPATH を次のように設定してください。

csch の場合

```
sun% setenv NLSPATH /usr/dt/lib/nls/msg/%L/%N.cat:/usr/dt/lib/nls/msg/C/%N.cat
```

sh (または ksh) の場合

```
sun$ NLSPATH=/usr/dt/lib/nls/msg/%L/%N.cat:/usr/dt/lib/nls/msg/C/%N.cat  
sun$ export NLSPATH
```

ボリューム管理ユーザーインターフェース

ボリューム管理用ユーザーインターフェース eject_popup および volmissing_popup のラベルとメッセージが日本語で表示されます。

日本語環境固有のアプリケーション

日本語環境をより使いやすくするために、次のプログラムが提供されています。

cs00 ユーザー辞書ツール

cs00 ユーザー辞書ツール (udicmtool) は cs00 を使用する際、ユーザー辞書に対して、登録されていない単語を登録したり、不要な単語を削除したりするためのツールです。また、辞書形式ファイルを編集してユーザー専用の辞書も作成できます。辞書登録ツールの「利用中の辞書」モードで変更された内容は、その時点で動作しているアプリケーションに、すぐに反映できます。

使用方法などについては、『cs00 ユーザーズガイド』と udicmtool(1) のマニュアルページを参照してください。

ワークスペースプロパティを使った環境の切り替え

日本語 OpenWindows のワークスペースプロパティの機能を使用して、動作環境を日本語から英語、または英語から日本語に切り替えることができます。

詳細は、第 12 章を参照してください。

日本語オンラインヘルプ

日本語 Magnify Help

ポインタを適当な場所に置いてキーボードの左側の Help キーを押すと、その時にポインタが置かれている場所に応じて、図 10-1 のようなヘルプメッセージを表示するウィンドウが表示されます。このウィンドウ内には日本語でメッセージが表示されます。

ワークスペースプロパティで、カテゴリの「ロケール」と「表示ロケール」を切り替えると、表示されるメッセージが日本語と英語で切り替わります。

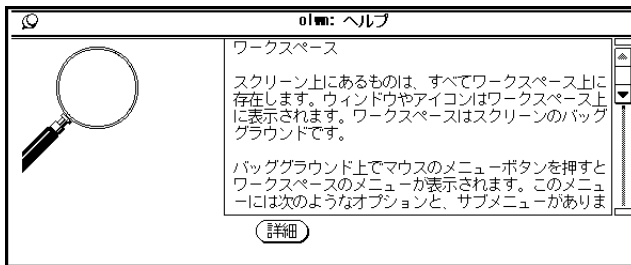


図 10-1 日本語 Magnify Help フォントグループの追加」 ウィンドウ

日本語ヘルプハンドブック

Magnify Help™ ウィンドウの「詳細」ボタンをクリックするか、ワークスペースメニューから「ヘルプ」を選択すると、日本語ヘルプハンドブックがヘルプビューワに表示されます。

日本語ヘルプハンドブックとヘルプビューワの詳細は、『OpenWindows ユーザーズガイド』を参照してください。

日本語オンラインマニュアル

環境変数 LANG または LC_MESSAGES が ja に設定されている場合、man コマンドで日本語のオンラインマニュアルを参照できます。

また、オンラインマニュアルには日本語と英語の両方があり、OpenWindows の各 DeskSet と日本語環境用プログラムの他、OpenWindows が提供する各種のコマンドについての説明を man コマンドで表示できます。

日本語フォントの使い方

XView アプリケーション

DeskSet などの XView のアプリケーションから、いろいろな種類の日本語フォントを指定して使用できます。たとえば、大きさ 24 のゴシック体ボールドのフォントを使用するには、次のように指定します。

```
sun% shelltool -Wt gotb-24
```

got はゴシック、b はボールド、数字は大きさをそれぞれ表します。また、明朝は min、ミディアムは m を表わすので、組み合わせて使用できます。特に指定がない場合には、デフォルトのフォントとして gotm-14 が使用されます。

次に、XView アプリケーション起動時にコマンドオプションとして指定できるフォントをまとめます。

表 10-1 XView のコマンドオプションフォント

gotb-12、14、16、20、24

gotm-12、14、16、20、24

minb-12、14、16、20、24

表 10-1 XView のコマンドオプションフォント 続く

minm-12、14、16、20、24

minm-12、14、16、20、24

OLIT アプリケーション

OLIT ツールキットを使用したアプリケーションを起動すると、デフォルトフォントとして各 EUC コードセットに対して次のフォントが使用されます。

表 10-2 OLIT のデフォルトフォント 1

コードセット 0	-sun-gothic-medium-r-normal-16-140-75-75-c-70-jisx0201.1976-0
----------	---

コードセット 1	-sun-gothic-medium-r-normal-16-140-75-75-c-140-jisx0208.1983-0
----------	--

コードセット 2	-sun-gothic-medium-r-normal-16-140-75-75-c-70-jisx0201.1976-0
----------	---

デフォルトフォントを変更するには、`olDefaultFont` や他の適切なフォントリソースに対して XLFD フォント名を直接指定してください。次に、リソースファイル内で、`olDefaultFont` にフォント名を設定する例を示します。

```
*olDefaultFont: -sun-gothic-bold-r-normal--26-240-75-75-c-120-jisx0201.1976-0,\  
-sun-gothic-bold-r-normal--26-240-75-75-c-240-jisx0208.1983-0,\  
-sun-gothic-bold-r-normal--26-240-75-75-c-120-jisx0201.1976-0
```

この設定により、各 EUC コードセットに対して次のフォントが使用されます。

表 10-3 OLIT のデフォルトフォント 2

コードセット 0	-sun-gothic-bold-r-normal-26-240-75-75-c-120-jisx0201.1976-0
----------	--

コードセット 1	-sun-gothic-bold-r-normal-26-240-75-75-c-240-jisx0208.1983-0
----------	--

コードセット 2	-sun-gothic-bold-r-normal-26-240-75-75-c-120-jisx0201.1976-0
----------	--

OLIT のアプリケーション起動時にフォントを指定する場合は、フォント名として XLFD 名を使用します。次の例を参考にしてください。

```
sun% olitsampler -fn \  
"-misc-fixed-medium-r-normal--14-130-75-75-c-70-iso8859-1, \  
-misc-fixed-medium-r-normal--14-130-75-75-c-140-jisx0208.1983-0, \  
-misc-fixed-medium-r-normal--14-130-75-75-c-70-jisx0201.1976-0"
```

日本語対応ツールキットによる日本語プログラムの開発環境

XView

国際化された XView ライブラリを使用して、日本語のウィンドウアプリケーションを開発できます。こうして開発したアプリケーションを日本語 OpenWindows 環境で使用することによって、日本語の入力表示など日本語を取り扱うことが可能になります。

国際化された XView ライブラリを使用してコンパイルする場合には、「-DOW_I18N」を明示的に指定します。

また、アプリケーション実行時に使用されるライブラリを指定するために、次のように「-R/usr/openwin/lib」も明示的に指定してください。

```
sun% cc -DOW_I18N -I/usr/openwin/include -L/usr/openwin/lib \  
-R/usr/openwin/lib -lxview -lX11 ...
```

国際化された XView の詳細は、『XView Developer's Notes』を参照してください。

OLIT

国際化された OLIT ライブラリを使用して、日本語のウィンドウアプリケーションを開発できます。こうして開発したアプリケーションを日本語 OpenWindows 環境で使用することによって、日本語の入力や表示などを取り扱うことが可能になります。

国際化された OLIT ライブラリを使用してコンパイルするには、次のように行います。

```
sun% cc -I/usr/openwin/include -R/usr/openwin/lib -L/usr/openwin/lib \  
-lXo1 -lXt -lX11 ...
```

日本語入力をともなうアプリケーション

TextEdit、TextField、ScrollingList、TextLine の各ウィジェットを使用した日本語入力をともなうアプリケーションを起動する際は、次の点に注意してください。

- 日本語入力サーバー `htt` または `atok8` が起動されていること

`htt` については第 8 章の98ページの「日本語入力サーバー(htt)」、`atok8` については『ATOK8 ユーザーズガイド』を参照してください。

- `XtNimPreeditStyle` および `XtNimStatusStyle` リソースに適切な値が設定されていること

リソースファイル内で、`XtNimPreeditStyle` および `XtNimStatusStyle` に値を設定する例を次に示します。

```
*imPreeditStyle:    onTheSpot  
*imStatusStyle:     imDisplaysInClient
```

これにより、`XtNimPreeditStyle` が `OL_ON_THE_SPOT` に、`XtNimStatusStyle` が `OL_IM_DISPLAYS_IN_CLIENT` に設定されます。

国際化された OLIT の詳細は、『OLIT リファレンスマニュアル』を参照してください。

日本語 OpenWindows の環境設定

この章では、JFP 上で日本語環境を使用するための初期設定の方法、日本語 OpenWindows の起動方法、さらに日本語 OpenWindows 2.0.1 環境からの移行方法と、日本語 OpenWindows 3.2 以前の環境からの移行方法について説明します。

初期設定と起動

ユーザーが JFP の日本語環境を使用する場合は、サーバーを起動する前に次の作業を行う必要があります。

1. 次のように環境変数 **LANG** を「**ja**」に設定します。

```
sun% setenv LANG ja
```

注 - Solaris では、通常、システム起動時にログインサーバー (dtlogin) が起動します。環境変数 LANG を変更するには、ログインサーバーの「オプション」メニューからロケールを設定します。詳細は、第 7 章を参照してください。

2. **JLE 1.1.2** 以前のシステムと共存するネットワーク上では、次のように環境変数 **LANG** を「**japanese**」に設定します。

```
sun% setenv LANG japanese
```

3. 日本語 OpenWindows を起動します。

注 - Solaris では、通常、システム起動時にログインサーバー (dtlogin) が起動します。日本語 OpenWindows を起動する場合は、ログインサーバーの「オプション」メニューの「セッション」から「OpenWindows デスクトップ」を選択してログインします。詳細は、第7章を参照してください。

一般的な設定に関しては、openwin シェルスクリプトがデフォルトのサーバー構成に関するすべてを処理します。openwin シェルスクリプトは、次のように実行します。

```
sun% /usr/openwin/bin/openwin
```

注 - LANG を「C」に設定した場合または何も設定していない場合は、日本語 OpenWindows が英語環境で起動されます。日本語 OpenWindows を終了せずに、ワークスペースプロパティを使用して英語環境を日本語環境に切り替える方法は、現時点ではサポートされていません。

日本語 OpenWindows 2.0.1 環境からの移行

カスタマイズしている日本語 OpenWindows 2.0.1 システム環境を日本語 OpenWindows 3.x システムに移行する場合、またはネットワーク上でホームディレクトリが JLE システムと共有されていて、日本語 OpenWindows 2.0.1 がそのホームディレクトリ上で起動されていた場合に

は、\$HOME/.xinitrc、\$HOME/.Xdefault、\$HOME/.openwin-menuなどを設定し直す必要があります。各ファイルの設定方法については、『OpenWindows ユーザーズガイド (上級編)』を参照してください。

キーの機能定義

日本語 OpenWindows 3.x システムではマウスレスオペレーション機能を実現するため、日本語 OpenWindows 2.0.1 のキーのいくつかの機能が他のキーに定義されています。日本語 OpenWindows 3.x で、これらのキーファンクションを日本語 OpenWindows 2.0.1 の定義に戻すには、`.Xdefaults` に次の記述を追加します。

```
OpenWindows.SunViewKeys: true
OpenWindows.Mouseless: false
OpenWindows.KeyboardCommand.DataStart: Return+Shift+Ctrl,R7+Ctrl
OpenWindows.KeyboardCommand.DataEnd: Return+Ctrl,End+Ctrl,R13+Ctrl
OpenWindows.KeyboardCommand.Translate: R2+Meta
```

日本語 OpenWindows 3.2 以前の環境からの移行

デフォルトのワークスペースメニューのファイル名とパス名が、次のように変更されています。

変更前

```
/usr/openwin/lib/openwin-menu.locale
/usr/openwin/lib/openwin-menu-programs.locale
/usr/openwin/lib/openwin-menu-utilities.locale
/usr/openwin/lib/openwin-menu-demo.locale
/usr/openwin/lib/openwin-menu-demo-fulldemo.locale
/usr/openwin/lib/openwin-menu-demo-nodemo.locale
```

変更後

```
/usr/openwin/share/locale/locale/olwm/openwin-menu
/usr/openwin/share/locale/locale/olwm/openwin-menu-programs
/usr/openwin/share/locale/locale/olwm/openwin-menu-utilities
/usr/openwin/share/locale/locale/olwm/openwin-menu-demo
/usr/openwin/share/locale/locale/olwm/openwin-menu-demo-fulldemo
/usr/openwin/share/locale/locale/olwm/openwin-menu-demo-nodemo
```

ホームディレクトリの `.openwin-menu` ファイルなどで変更前のファイル名やパス名を使用している場合は、変更する必要があります。特に、`INCLUDE` や `MENU` などのキーワードを使用して、次の例のように変更前のファイル名を指定している場合は、そのメニューが表示されなくなります。

例

```
"プログラム" INCLUDE openwin-menu-programs
```

注 - システムは次の優先順位でファイルを参照します。

1. \$HOME/.openwin-menu-programs
 2. /usr/openwin/share/locale/\$LANG/olwm/openwin-menu-programs
-

その他、各環境設定ファイルの詳細は、『OpenWindows ユーザーズガイド (上級編)』を参照してください。

日本語 OpenWindows のカスタマイズ

この章では、日本語 OpenWindows の日本語環境をカスタマイズする方法について説明します。OpenWindows 全般に対する環境のカスタマイズについては、『OpenWindows ユーザーズガイド』を参照してください。

ワークスペース

日本語 OpenWindows のワークスペースプロパティ機能を使用して、動作環境を日本語と英語とで切り替えることができます。

ワークスペースメニューから「プロパティ」を選択すると、図 12-1に示すような「ワークスペース・プロパティ」ポップアップウィンドウが表示されます。

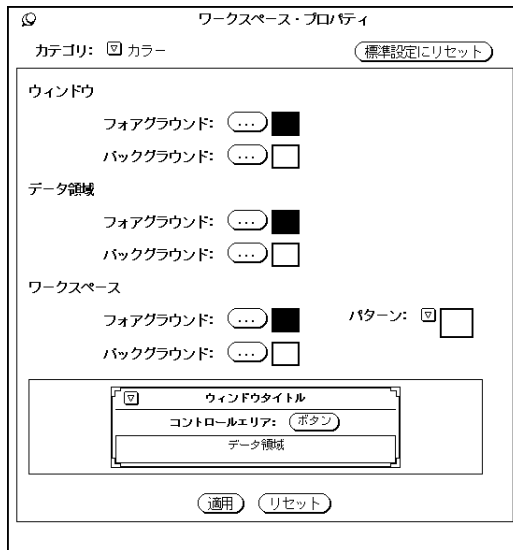


図 12-1 起動時の「ワークスペース・プロパティ」ウィンドウ

次に、「カテゴリ」から「ロケール」を選択すると、図 12-2 に示す項目が表示されます。

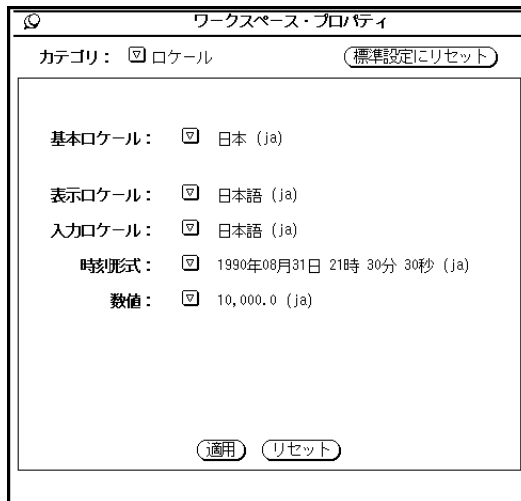


図 12-2 ロケールメニュー (ワークスペース・プロパティ)

基本ロケール

基本ロケールは、全体的な動作環境を日本語または英語のどちらにするかを選択します。「日本」または「(日本)」にすると、日本語表示や日本語入力ができます。「米国」にすると日本語の入出力はできません。

基本ロケールで「日本」を選択すると、自動的にその他の設定も日本語用になります。「米国」を選択すると、自動的にその他の設定も英語用になります。

ネットワーク上の JLE 1.1.2 以前のシステムとの互換性を持たせるために、ロケールを「japanese」に設定したい場合は、基本ロケールで「(日本)」を選択します。

注 - 基本ロケールを「日本」にして、次に述べるその他の設定を日本語または英語にすることはできますが、基本ロケールを「米国」にして、その他の設定を日本語にすることはできません。基本ロケールを「(日本)」にすると、その他の設定は自動的に「japanese」になります。その他の設定を英語にすることもできますが、「ja」にすることはできません。

その他の設定

表示ロケール

その他の設定の表示ロケールは、メニュー、ボタン、ヘルプメッセージなどを日本語または英語のどちらで表示するかを選択します。

たとえば、設定を「日本語」から「英語」に切り替えた後、「適用」をクリックすると、ワークスペースメニューは英語で表示されるようになり、次から起動するアプリケーションのメニューやボタンなどは英語になります。

入力ロケール

入力ロケールは、アプリケーションで日本語入力をできるようにするかどうかを選択します。

設定を「日本語」にすると、アプリケーションを起動したときに日本語入力ができるようになります。設定を「英語」にすると、日本語入力はできません。

時刻形式

時刻形式は、アプリケーションが時刻を表示するときに、日本式に「??年??月??日??時??分??秒」で表示するか、米国式に「??/??/?? ??:?:??」で表示するかを選択します。

数値

数値形式は、数字をどのような形式で表示するかを選択します。

ロケールの再定義

JFP は、動作環境を切り替えるためのロケールとして、英語用に「C」、日本語用に「ja」、互換性のために「japanese」を提供しています。

現時点では、ユーザーはロケール名を新しく定義できません。

EUC コードセット 3 (JIS X 0212)

日本語 OpenWindows では、日本語ロケールの文字集合として、コードセット 0、1、2 に加えて、コードセット 3 (JIS X 0212-1990 補助漢字) をサポートします。

日本語 OpenWindows では、日本語ロケール (ja、japanese) の EUC コードセット 0、1、2、3 に以下の文字集合をサポートしています。

- EUC コードセット 0 JIS X 0201-1976
- EUC コードセット 1 JIS X 0208-1990
- EUC コードセット 2 JIS X 0201-1976
- EUC コードセット 3 JIS X 0212-1990

仮想 LED

仮想 LED は、キーボードのロックキーの状態を画面上で確認するためのツールです。

日本語 (タイプ 4、タイプ 5) キーボードにはアルファベットとかなの入力を切り替えるロックキーが用意されていますが、キーボード本体にはその状態を示す LED は用意されていません。

仮想 LED は、「かなロック」の状態を画面上で確認できます。

起動方法

仮想 LED を起動するには、ワークスペースメニューから仮想 LED を選択するか、次のようにシェルツールまたはコマンドツールからコマンドを入力します。

```
sun% /usr/openwin/bin/vled
```

デフォルトの場合、仮想 LED は日本語キーボードを使用しているシステムでだけ起動されるように設定されています。

日本語キーボードを使用していないシステムで仮想 LED を使用する場合は、次のように `-unconditional` オプションを指定してください。

```
sun% /usr/openwin/bin/vled -unconditional
```

デフォルトでは「かなロック」、「Caps ロック」、「Num ロック」を示す状態で起動します。



図 13-1 vled 起動時の表示

オプション

仮想 LED には次のオプションがあります。

-unconditional

キーボードが日本語キーボードであるかどうかに関わらず、仮想 LED を起動できます。

注 - このオプションを使用して、日本語キーボード以外を使用するシステムで仮想 LED を起動した場合、Alt-Graph キーを押すと「かなロック」の LED が光りますが、実際にはかな入力はできません。

-kanaonly

「かなロック」の仮想 LED だけが表示されます。



図 13-2 vled -kanaonly の表示

-Wi (iconic)

仮想 LED がアイコンの状態で起動されます。



図 13-3 仮想 LED のアイコン

-display host:*x.x*

ディスプレイを指定できます。

-Ws *width height*

仮想 LED の起動時の大きさを指定できます。

-Wp *x y*

仮想 LED の起動時の表示位置を指定できます。

-WP *x y*

仮想 LED のアイコンの表示位置を指定できます。

-geometry (WxH)({+|-}X{+|-}Y)

仮想 LED の起動時の大きさと表示位置を指定できます。

-icon_geometry (WxH)({+|-}X{+|-}Y)

仮想 LED のアイコンの大きさと表示位置を指定できます。

-fg (-fourground) #rrggbb | *color_name*

仮想 LED で使用される文字と枠線の色を指定できます。

-bg (-background) #rrggbb | *color_name*

仮想 LED 内のバックグラウンドの色を指定できます。

-brightcolor #rrggbb | *color_name*

各キーのオン状態の色を指定できます。

-darkcolor #rrggbb | *color_name*

各キーのオフ状態の色を指定できます。

-intervaltime *usec*

キーのオン・オフの状態を検索する時間の間隔を指定できます。

ユーザー定義文字の登録 (Solaris 外字ツール)

Solaris 2.6 から、ユーザー定義文字は既存のフォントファイルに追加せずに、別フォントファイルとして編集し、既存のフォントと組み合わせて利用するようになりました。ユーザー定義文字を登録できる領域は次のとおりです。

表 14-1 ユーザー定義文字の登録可能領域

文字	登録可能な領域	文字セットの区域
日本語 EUC	0xf5a1 - 0xfefe	JIS X 0208 85 区 - 94 区
	0x8ff5a1 - 0x8ffefe	JIS X 0212 85 区 - 94 区
PCK	0xf040 - 0xf9fc	JIS X 0208 95 区 - 114 区
UTF-8	0xee8080 - 0xefa3bf (U+E000 - U+F8FF)	

このため、これまでのビットマップ用の「フォントエディタ」(fontedit)、アウトライン用の「タイプ 3 クリエータ」(type3creator) と「フォントマネージャ」(fontmanager) に代わり、Solaris 2.6 オペレーティング環境およびその互換バージョンの Solaris CDE 上で新たにユーザー定義文字を作成するツールとして、「Solaris 外字ツール」(sdtudctool) が提供されます。

「Solaris 外字ツール」は、日本語 EUC、PCK および UTF-8 に共通したユーザー定義文字を作成します。そのため、ユーザー定義文字を作成する場合、いずれかのロケールで作成するといずれのロケールでも利用できるようになります。

なお、フォントエディタ、タイプ3クリエータ、およびフォントマネージャを使って作成したユーザー定義文字を再利用する場合は、はじめに移行作業が必要です。詳細は、第 15 章を参照してください。

機能概要

「Solaris 外字ツール」には次の機能があります。

- ユーザー定義文字を Solaris 2.6 オペレーティング環境およびその互換バージョンでサポートするユーザー定義文字フォントファイル形式に保存する
- ビットマップのユーザー定義文字とアウトラインのユーザー定義文字を同時に作成する
- 必要なサイズのビットマップのユーザー定義文字を一度に作成する
- 別のフォントを参照しながら編集する
- ドラッグ&ドロップによりコードポイントの移動を簡単に行う
- BDF/PCF フォントファイルを編集する
- ユーザー定義文字の読みを辞書に登録するための、ユーザー定義文字辞書登録用中間ユーティリティを起動する

なお、作成されたユーザー定義文字フォントファイルは、日本語 OpenWindows 上でも共通して利用できます。

注 - 「Solaris 外字ツール」は Solaris CDE 上でのみサポートされます。日本語 OpenWindows 上でユーザー定義文字を利用する場合でも、Solaris CDE 環境上でユーザー定義文字を登録してください。

「Solaris 外字ツール」は、次の 3 つのウィンドウで構成されています。各ウィンドウ上のメニューの詳細は、136ページの「各機能の説明」を参照してください。

文字エディタ

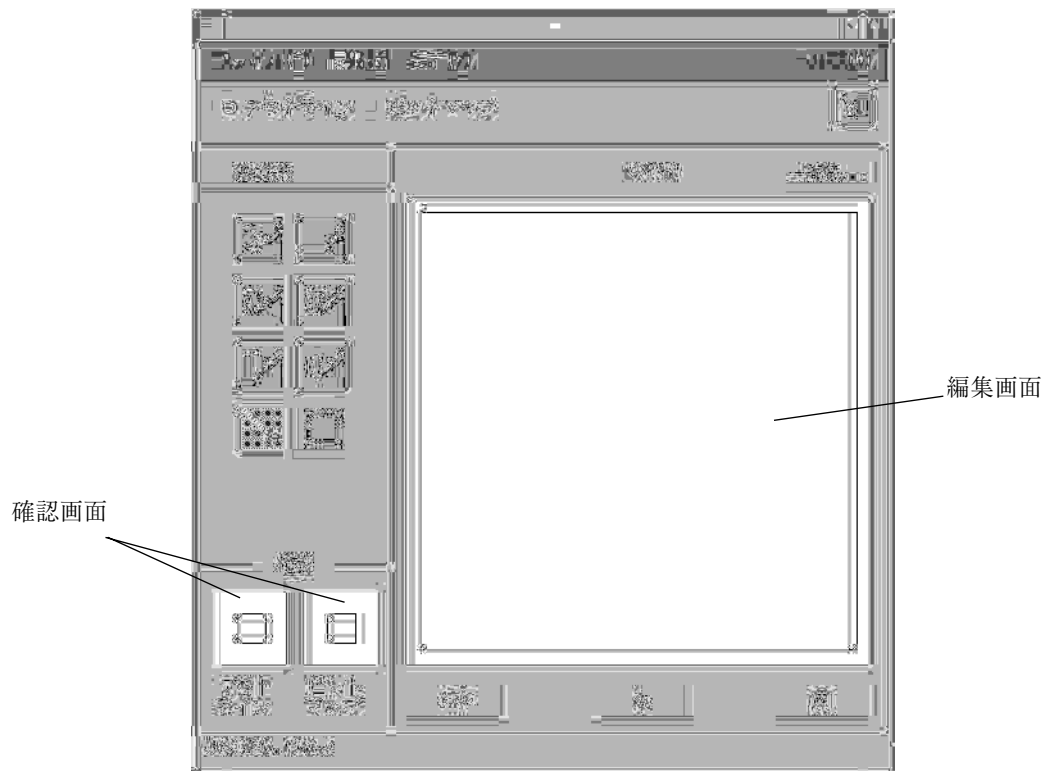


図 14-1 文字エディタウィンドウ (Solaris 外字ツール)

編集画面

ユーザー定義文字を編集する画面です。

参照画面

ユーザー定義文字を作成するときに参考にする文字を表示する画面です。「表示」メニューから「参照」、またはツールバーの右端にあるボタンを選択すると、編集画面の右に表示します。

確認画面

左側にアウトラインでの文字、右側にビットマップでの文字を表示します。

一覧表



図 14-2 一覧表ウィンドウ (Solaris 外字ツール)

「一覧表...」ボタンを選択するとこのウィンドウを表示し、現在登録しているユーザー定義文字を表示します。

参照表



図 14-3 参照表ウィンドウ (Solaris 外字ツール)

参照画面を表示したあと、「参照表...」ボタンを選択すると、このウィンドウを表示します。ユーザー定義文字を編集する場合に参考にしたい文字があるときに使用します。

起動方法

「Solaris 外字ツール」を起動するには、アプリケーションマネージャの「デスクトップアプリケーション」を開いて「Solaris 外字ツール」を選択するか、端末エミュレータで次のコマンドを入力します。



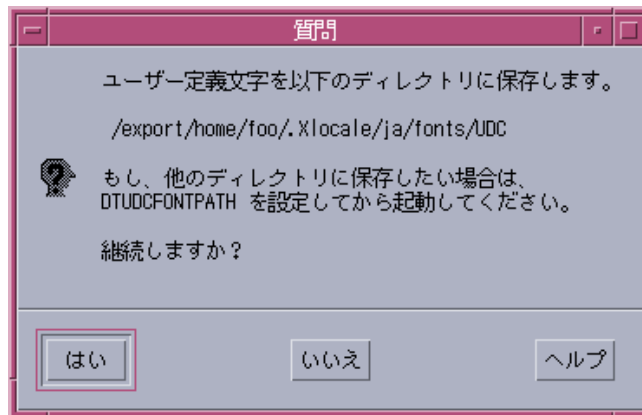
```
sun% /usr/dt/bin/sdtudctool
```

環境変数 DTUDCFONTPATH でユーザー定義文字を保存するディレクトリを指定していない場合は、以下のディレクトリに保存します。

ローカルユーザーの場合: \$HOME/.Xlocale/ja/fonts/UDC
スーパーユーザーの場合: \$OPENWINHOME/lib/locale/ja/X11/fonts/UDC

環境の設定にもよりますが、ローカルユーザーがユーザー定義文字を登録した場合は、登録者の環境でユーザー定義文字の利用が可能となります。また、スーパーユーザーがユーザー定義文字を登録した場合、登録したマシンおよび \$OPENWINHOME を共有するマシンでユーザー定義文字の利用が可能となります。また、各ユーザー間で環境変数 DTUDCFONTPATH に共有のディレクトリを指定することで、共有のユーザー定義文字を利用することもできます。

ユーザー定義文字を保存するディレクトリが存在しない場合は、起動時に次のダイアログが表示されます。

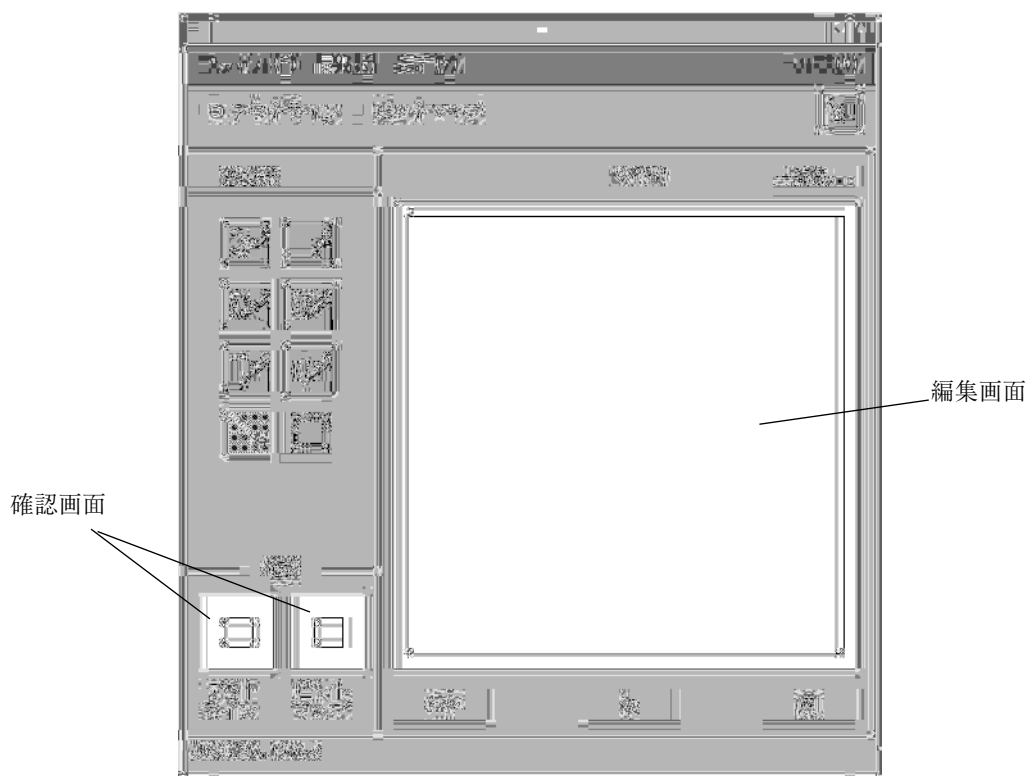


各機能の説明

「Solaris 外字ツール」で提供する機能について説明します。

文字エディタ

「Solaris 外字ツール」の起動時に表示します。



文字エディタのウィンドウタイトルには、ユーザー定義文字を読み込んだ場合にはその保存ディレクトリ名が、フォントファイルを読み込んだ場合にはファイル名を表示します。

描画ツール

描画ツールが提供する各ツールの機能と操作方法を表 14-2 に示します。描画ツールの中から 1 つのツールを選択すると、別のツールが選択されるまでそのツールが選択されています。

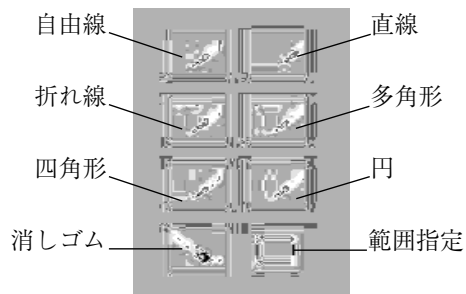


図 14-4 描画ツール (Solaris 外字ツール)

表 14-2 描画ツールの機能と操作方法

ツール名	機能	操作方法
自由線	手書きの線と個別のピクセルを描きます。	<ol style="list-style-type: none"> 1. 「描画ツール」メニューから「自由線」を選択します。 2. 描画面面上で左マウスボタンを押しながら描画します。 3. 左マウスボタンを離します。
直線	直線を描きます。	<ol style="list-style-type: none"> 1. 「描画ツール」メニューから「直線」を選択します。 2. 描画面面上で左マウスボタンを押しながら描画します。 3. 左マウスボタンを離します。
折れ線	連続する複数の線分を描きます。	<ol style="list-style-type: none"> 1. 「描画ツール」メニューから「折れ線」を選択します。 2. 描画面面上で、線分の開始地点と終了地点ごとに左マウスボタンをクリックします。クリックした場所から新たに線分が描画されます。 3. 線分の最終地点で左マウスボタンをダブルクリックします。

表 14-2 描画ツールの機能と操作方法 続く

ツール名	機能	操作方法
多角形	最初の描画線と最後の描画線の間を閉じた多角形を描きます。	<ol style="list-style-type: none"> 1. 「描画ツール」メニューから「多角形」を選択します。 2. 描画面面上で、描画線の開始地点と終了地点ごとに左マウスボタンをクリックします。 クリックした場所から新たに線分が描画されます。 3. 線分の最終地点で左マウスボタンをダブルクリックします。
四角形	四角形を描きます。	<ol style="list-style-type: none"> 1. 「描画ツール」メニューから「四角形」を選択します。 2. 描画面面上で左マウスボタンを押しながら描画します。 3. 左マウスボタンを離します。
円	円を描きます。	<ol style="list-style-type: none"> 1. 「描画ツール」メニューから「円」を選択します。 2. 描画面面上で左マウスボタンを押しながら描画します。 3. 左マウスボタンを離します。
消しゴム	<p>指定されたピクセルを消去します。</p> <p>注: アウトラインモードでは利用できません。「範囲指定」で範囲を指定し、「編集」メニューを利用し消します。</p>	<ol style="list-style-type: none"> 1. 「描画ツール」メニューから「消しゴム」を選択します。 2. 描画面面上で左マウスボタンを押しながら消します。
範囲指定	<p>範囲を指定します。「編集」メニューのコマンドを使う場合は、最初に「範囲指定」で範囲を指定する必要があります。なお、指定領域の移動は、指定領域の中心付近を、左マウスボタンで押しながら移動します。</p>	<ol style="list-style-type: none"> 1. 「描画ツール」メニューから「範囲指定」を選択します。 2. 描画面面上で左マウスボタンを押しながら範囲を指定します。 3. 左マウスボタンを離します。

メニュー

文字エディタには次のメニューがあります。

ファイル

■ ユーザー定義文字

ユーザー定義文字を読み込みます。環境変数 DTUDCFONTPATH が無効な場合は、\$HOME/.Xlocale/ja/fonts/UDC から読み込みます。

■ 開く...



フォントファイルを指定して読み込む場合に使用します。次のファイル形式のフォントファイルを指定できます。

- BDF 形式のフォントファイル
- PCF 形式のフォントファイル

■ 保存

- ユーザー定義文字を読み込んだ場合

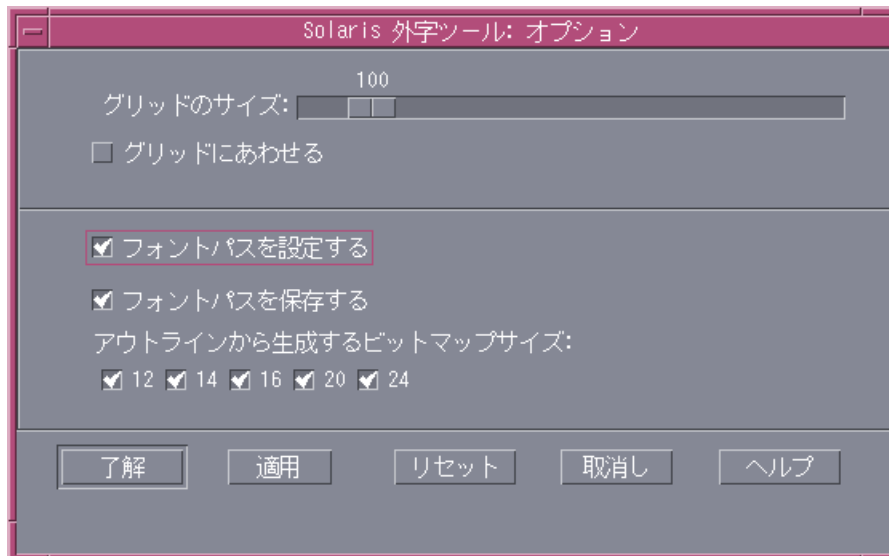
ユーザー定義文字を別フォントファイルとして保存し、利用するために必要なファイルを設定します。その後、ユーザー定義文字辞書登録用中間ユーティリティ (sdtudc_register) を起動します。

- フォントファイルを直接指定して読み込んだ場合
読み込んだフォントファイルに保存します。
- 別名保存...



編集した結果を別名のフォントファイルとして保存する場合に利用します。ユーザー定義文字を読み込んだ場合は選択できません。なお、次のいずれかの拡張子を指定する必要があります。

- BDF 形式のフォントとして保存する場合 .bdf
- PCF 形式のフォントとして保存する場合 .pcf
- オプション



- グリッドのサイズ

アウトライン編集モード時のグリッドのサイズを指定します。単位はポイントです。

- グリッドに合わせる

アウトライン編集モード時に描画位置をグリッド上に合わせる指定をします。

- フォントパスを設定する

現行セッションのサーバーのフォントパスにユーザー定義文字を保存したディレクトリを追加します。このチェックボックスが選択されていない場合、フォントパスへの追加は行われず、登録したユーザー定義文字を利用できません。

なお、この項目を選択しないで、「Solaris 外字ツール」終了後にサーバーのフォントパスに追加する場合は、次のようにしてください。

ローカルユーザーの場合

1. DTUDCFONTPATH を環境変数として設定している場合

```
sun% $OPENWINHOME/bin/xset +fp $DTUDCFONTPATH/Bitmaps, \
$DTUDCFONTPATH/Type1, $DTUDCFONTPATH/CID
```

2. DTUDCFONTPATH を環境変数として設定していない場合

```
sun% $OPENWINHOME/bin/xset fp+ $OPENWINHOME/lib\  
/locale/ja/fonts/UDC/Bitmaps,\  
$OPENWINHOME/lib/locale/ja/fonts/UDC/Type1,\  
$OPENWINHOME/lib/locale/ja/fonts/UDC/CID
```

なお、指定したフォントパスが正しく設定されているかの確認は、以下のように `xset` を実行してください。

```
sun% xset q
```

詳細は、`xset (1)` のマニュアルページを参照してください。

- フォントパスを保存する

サーバーのフォント指定ファイルへユーザー定義文字を保存したディレクトリを追加します。これにより、以降サーバーが立ち上がる際、自動的にフォントパスにユーザー定義文字を保存したディレクトリを追加します。フォント指定ファイルは、以下の通りです。選択されていない場合は追加されません。

ローカルユーザーがユーザー定義文字を保存した場合のフォント指定ファイル

```
$HOME/.OWfontpath
```

スーパーユーザーがユーザー定義文字を保存した場合のフォント指定ファイル

```
$OPENWINHOME/$LANG/OWfontpath
```

この項目を選択しないで、「Solaris 外字ツール」終了後にフォント指定ファイルへ追加する場合は、`vi` などのエディタコマンドを使用して、上記のフォント指定ファイルを以下の様に編集してください。

ローカルユーザーがユーザー定義文字を保存した場合

```
sun% /usr/bin/vi $HOME/.OWfontpath
```

フォント指定ファイルを次のように編集します。

```
$OPENWINHOME/lib/locale/ja/X11/fonts/UDC/Bitmaps <-- 追加
$OPENWINHOME/lib/locale/ja/X11/fonts/UDC/Type1 <-- 追加
$OPENWINHOME/lib/locale/ja/X11/fonts/UDC/CID <-- 追加
```

スーパーユーザーがユーザー定義文字を保存した場合

```
sun# /usr/bin/vi $OPENWINHOME/lib/locale/$LANG/OWfontpath
```

フォント指定ファイルを次のように編集します。

```
$OPENWINHOME/lib/locale/ja/X11/fonts/75dpi
$OPENWINHOME/lib/locale/ja/X11/fonts/TT
$OPENWINHOME/lib/locale/ja/X11/fonts/TTbitmaps
$OPENWINHOME/lib/locale/ja/X11/fonts/UDC/Bitmaps <-- 追加
$OPENWINHOME/lib/locale/ja/X11/fonts/UDC/Type1 <-- 追加
$OPENWINHOME/lib/locale/ja/X11/fonts/UDC/CID <-- 追加
```

注 - 上記のインタフェースは変更される場合があります。フォント指定ファイルの編集は、ユーザー定義文字を利用する場合にだけ行なってください。

- アウトラインから生成するビットマップサイズ

ユーザー定義文字を保存する際、自動生成されるビットマップフォントのサイズを指定します。

- 終了

「Solaris 外字ツール」を終了します。

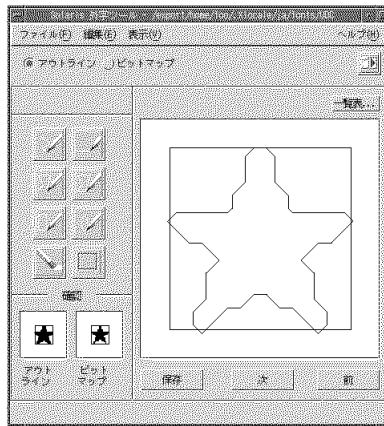
編集

モードの切り替えは、ツールバー上に表示されている「アウトライン」と「ビットマップ」ボタンを切り替えて指定します。

次の2つのモードがあります。

アウトライン ビットマップ

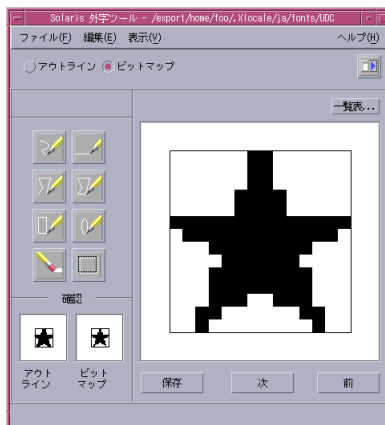
- アウトライン編集モード



文字の作成をアウトラインベースで行います。なお、実際のイメージはアウトラインで囲まれた領域が塗り込まれたものとなります。

「表示」→「塗り込み」でイメージを編集画面に表示できます。

■ ビットマップ編集モード



文字の作成をビットマップベースで行います。

「編集」メニューには次のボタンがあります。

■ 元に戻す

直前の状態に戻ります。

■ カット

「範囲指定」で指定した領域を切り取り、バッファに取り込みます。

■ コピー

「範囲指定」で指定した領域をバッファに取り込みます。次にペーストで、バッファの内容を編集画面に描くことができます。

- ペースト

バッファの内容を編集画面に描きます。

- 削除

「範囲指定」で指定した領域を削除します。

- 回転

アウトラインモード時に「範囲指定」で指定した領域を回転します。

- 斜め

アウトラインモード時に「範囲指定」で指定した領域を斜めの領域に変換します。

- 反転

ビットマップモード時に「範囲指定」で指定した領域の白黒を反転します。

注・アウトライン表示時の「範囲指定」では、文字を構成するオブジェクトのすべてのコントロールポイントが指定領域に含まれなければなりません。コントロールポイントの表示は「表示」→「コントロール」で行います。

表示

- 塗りつぶし

アウトライン表示の場合、アウトラインで囲まれた領域を塗りつぶします。実際にフォントとして表示されるのはこのイメージです。

- コントロール

アウトラインを描画する場合の基準点を表示します。アウトラインモードで範囲指定をする場合、すべてのコントロールポイントを指定領域に含んだ場合にだけ範囲指定は有効になります。

- ドラッグ

イメージ表示の移動（「回転」または「斜め」）途中の表示方法を指定します。

- グリッド

編集画面上にグリッドを表示します。

- 参照

参照画面を開きます。

ヘルプ

- 概要

「Solaris 外字ツール」のオンラインヘルプを表示します。

- ヘルプの使い方

AnswerBook を起動します。

- バージョン

「Solaris 外字ツール」のバージョン番号を表示します。

一覧表

一覧表は、ユーザー定義文字を読み込んだ場合はユーザー定義文字を、フォントファイルを読み込んだ場合はファイルに含まれる文字を表示します。一覧表上の文字を選択すると編集画面上に表示され、編集対象になります。



一覧表のウィンドウタイトルには、ユーザー定義文字を読み込んだ場合にはその保存ディレクトリ名が、フォントファイルを読み込んだ場合にはファイル名が表示されます。

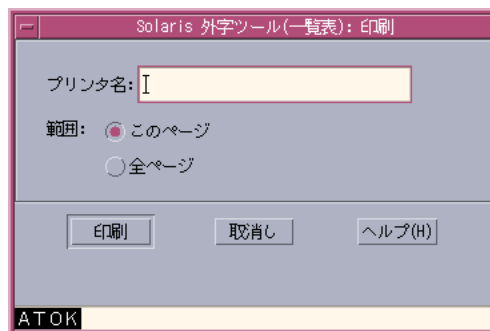
ツールバー上には、次のメニューが表示されます。

- ページ
スライダーを使って希望のページに移動します。
- 左矢印
前ページに移動します。
- 右矢印
次ページに移動します。

メニューバーには、次のメニューが表示されます。

ファイル

- 印刷...
印刷ダイアログを開きます。



- 閉じる
一覧表を閉じます。

編集

- 元に戻す
一つ前の編集状態に戻ります。
- カット
指定されている文字を削除し、バッファに格納します。
- コピー

指定されている文字をバッファに格納します。

- ペースト

バッファの内容を指定された場所に描画します。

- 削除

指定されている文字を削除します。

表示

- 次ページ

次のページがある場合、「次ページ」で1ページ進みます。

- 前ページ

前ページがある場合、「前ページ」で1ページ戻ります。

- サイズ

表示されているフォントのサイズを変更します。ただし、ビットマップフォントを読み込んでいる場合はサイズを変更できません。

- コード

表示されている文字のコード体系を変更します。

参照表

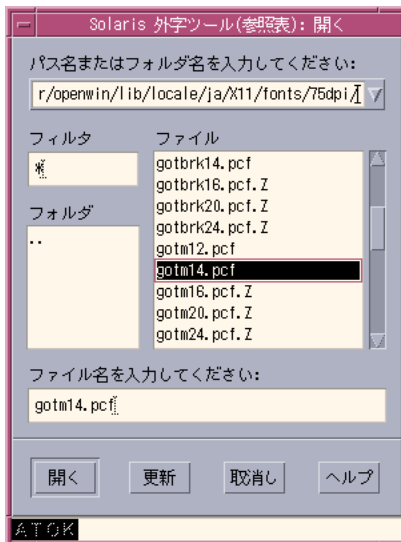
参照表は、「ファイル」メニューの内容を除き、一覧表とほとんど同じインタフェースを持っています。



参照表のウィンドウタイトルには、ユーザー定義文字を読み込んだ場合にはその保存ディレクトリ名が、フォントファイルを読み込んだ場合にはファイル名が表示されます。

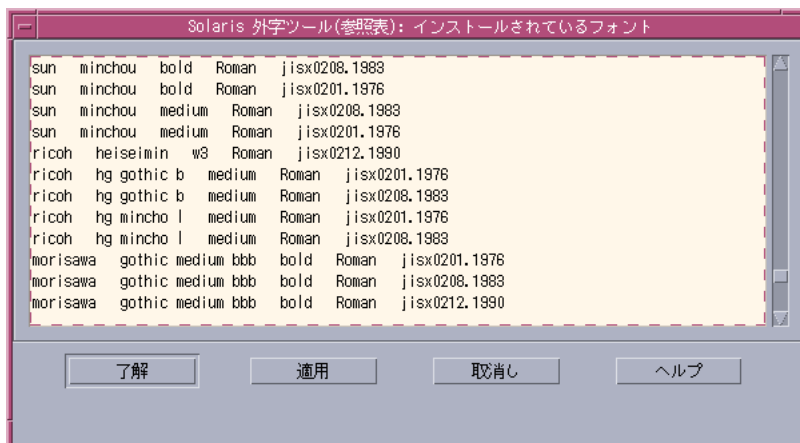
ファイル

- 開く



参照するフォントファイルを指定します。

- インストールされているフォントを選択



システムで利用可能なフォントの一覧表を表示します。フォントを選択すると参照表に表示します。

■ 閉じる

参照表を閉じます。

その他の機能

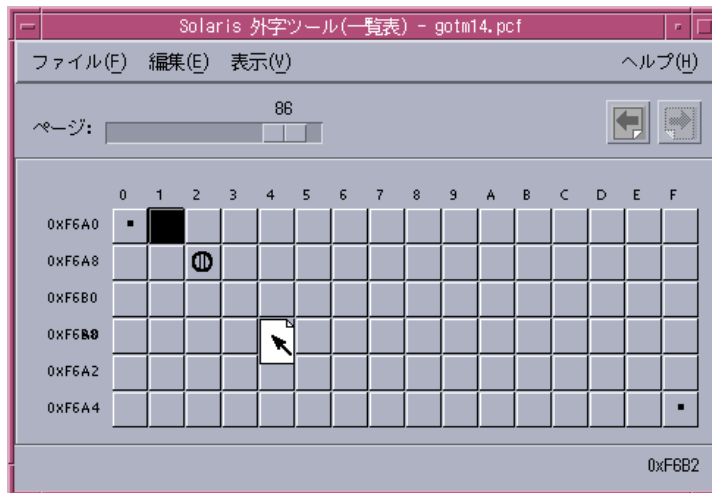
「Solaris 外字ツール」では、一覧表上でのドラッグ&ドロップ、一覧表から編集画面へのドラッグ&ドロップ、参照表から編集画面へのドラッグ&ドロップ、および参照表から一覧表へのドラッグ&ドロップをサポートしています。

ドラッグ&ドロップ機能は、文字のコードポイントの移動や、すでに登録されている文字を利用して文字を作成する場合に便利です。

コードポイントの移動例

コードポイントを移動する際の操作例を次に示します。

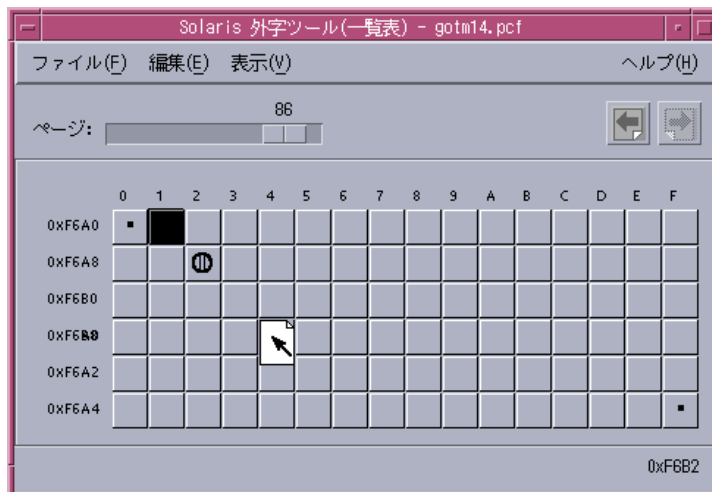
1. 移動したいコードポイントの文字をマウスのアジャスト (中央) ボタンを押して選択したまま、移動先のコードポイントへカーソルをドラッグ&ドロップします。



参照表からの文字のコピー例

次のいずれかの操作を実行すると文字をコピーできます。

- ◆ 参照表上のコピーしたい文字をマウスのアジャスト (中央) ボタンを押して選択したまま、一覧表のコピー先のコードポイントへカーソルをドラッグ&ドロップします。



- ◆ 一覧表上でコードポイントを指定した後、参照表上の参照したい文字をマウスのアジャスト (中央) ボタンを押して選択したまま、編集画面上にドラッグ&ドロップします。保存ボタンを選択します。

ユーザー定義文字の日本語入力システムへの登録方法

ユーザー定義文字を単語として日本語入力システムの辞書ファイルへ登録することにより、ユーザー定義文字の入力が容易になります。登録を行わない場合でも JIS 区点コード入力または、16 進コード入力によりユーザー定義文字を入力することが出来ます。ここでは、「Solaris 外字ツール」を利用して各日本語入力システムの辞書ファイル登録までの手順を説明します。単語登録に関する詳細は、「Wnn6 ユーザーズガイド」、「ATOK8 ユーザーズガイド」または「cs00 ユーザーズガイド」を参照して下さい。

1. 「**Solaris 外字ツール**」を起動します。詳細は、135ページの「起動方法」を参照してください。
2. 「描画モード」を選択し、編集画面上に文字を描画します (起動時には「自由線」が設定されています)。
3. 文字の描画が終了したら、編集画面下の「保存」ボタンを押します。
4. 「次」ボタンを押し、次のコードポイントに移動します。
5. すべてのユーザー定義文字の登録が終了したら、「ファイル」メニューから「保存」を選択します。

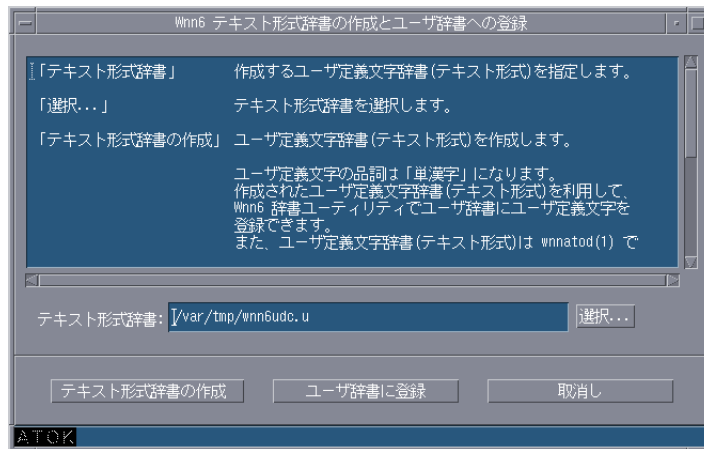
なお、「一覧表...」ボタンを押すと、登録したユーザー定義文字の一覧表が表示されます。この一覧表から直接、編集するユーザー定義文字を選択したり、登録するコードポイントを指定したりできます。詳細は、「一覧表」を参照してください。

「Solaris 外字ツール」は、ユーザー定義文字を保存したディレクトリを自動的に現行セッションのフォントパスに組み込みます。詳細は、136ページの「文字エディタ」、140ページの「ファイル」の「オプション」、および 160ページの「オプション」を参照してください。

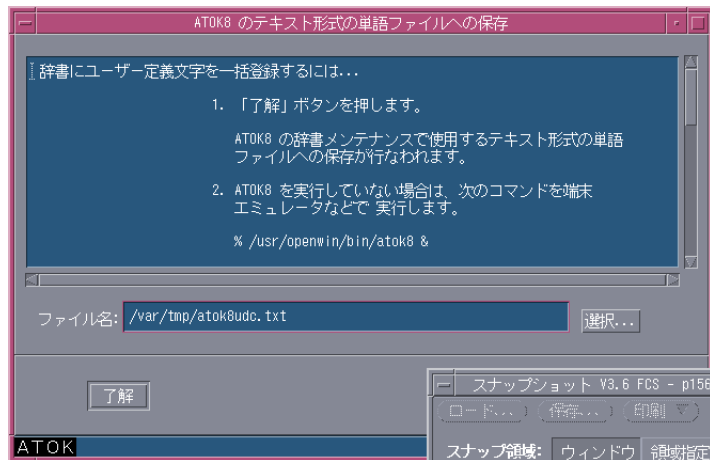
6. ユーザー定義文字の保存が終了すると、ユーザー定義文字の読みを辞書に登録するためのユーザー定義文字辞書登録用中間ユーティリティが自動的に起動されます。



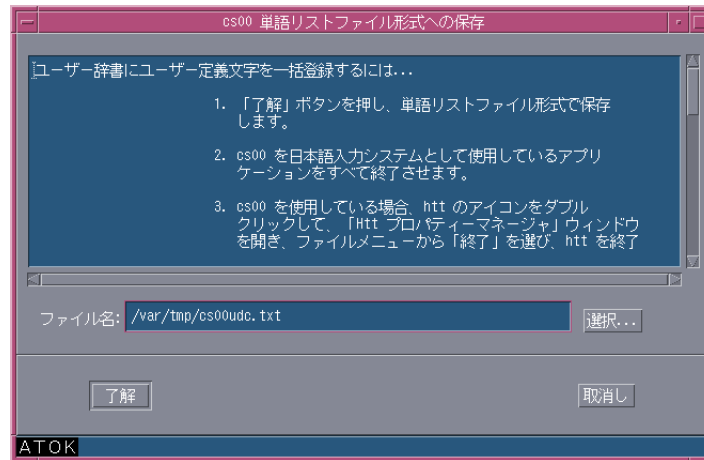
7. 単語登録対象の日本語入力システムを選択し(**Wnn6**、**ATOK8** または **cs00**)、単語登録したいユーザー定義文字の「読み」および「コメント」を入力し、「了解」ボタンまたは「適用」ボタンを押します。「コメント」は、**Wnn6** のみ辞書ファイルに登録されます。
8. 7. で選択した日本語入力システム別にウィンドウが表示されます。ウィンドウの説明に従って処理を続けてください。
 - Wnn6 テスト形式辞書への保存



■ atok8wordlist 形式ファイルへの保存



■ cs00 単語リストファイル形式への保存



なお、ユーザー定義文字の「読み」を辞書ファイルに登録しない場合、アプリケーションでの指定方法はコード入力となります。

ユーザー定義文字のアプリケーションでの利用方法

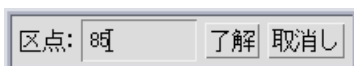
ユーザー定義文字の利用は、登録後に起動したアプリケーションから有効になります。

表 14-3 日本語入力システムで利用可能なユーザー定義文字のコード範囲

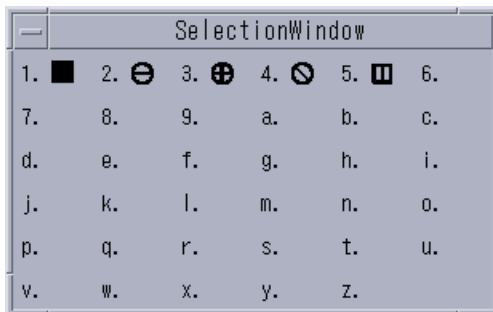
日本語入力システム	図形文字集合	コード範囲
Wnn6、cs00	日本語 EUC コードセット 1	0xf5a1 から 940 文字
	日本語 EUC コードセット 2	0x8ff5a1 から 940 文字
	PC 漢字(PCK)	0xF040 から 1880 文字
	UTF-8	0xee8080 から 1880 文字
ATOK8	日本語 EUC コードセット 1	0xf5a1 から 940 文字
	日本語 EUC コードセット 2	利用できません
	PC 漢字(PCK)	0xf040 から 940 文字
	UTF-8	0xee8080 から 940 文字

日本語入力システム Wnn6 を利用している場合

1. 区点入力ウィンドウを表示します。
2. ユーザー定義文字を登録したコードポイントを区点コードで指定します。
たとえば、85 区 1 点から一覧を表示して選択する場合は、区点入力ウィンドウで 85 を入力し、候補一覧を表示します。



一覧から選択します。



日本語入力システム **ATOK8** を利用している場合

1. **JIS** 入力モードにします。
2. ユーザー定義文字を登録したコードポイントを **JIS** コードで指定します。
たとえば、85 区 1 点が JIS 0x7521 なので、75 を入力し候補一覧を表示します。



日本語入力システム **cs00** を利用している場合

1. 区点入力モードにします。
2. ユーザー定義文字を登録したコードポイントを区点コードで指定します。
たとえば、85 区 1 点から一覧を表示して選択する場合は、85 を入力し候補一覧を表示します。



詳細は、『Wnn6 ユーザーズガイド』、『ATOK8 ユーザーズガイド』、または『cs00 ユーザーズガイド』を参照してください。

ユーザー定義文字の DPS 上での利用方法

「Solaris 外字ツール」で作成されたユーザー定義文字は、以下のフォントに組み込まれます。

JIS エンコーディングのフォント

GothicBBB-Medium-H	Ryumin-Light-H
GothicBBB-Medium-V	Ryumin-Light-V
GothicBBB-Medium-Hojo-H	Ryumin-Light-Hojo-H
GothicBBB-Medium-Hojo-V	Ryumin-Light-Hojo-V

EUC エンコーディングのフォント

GothicBBB-Medium-EUC-H	Ryumin-Light-EUC-H
GothicBBB-Medium-EUC-V	Ryumin-Light-EUC-V
GothicBBB-Medium-Hojo-EUC-H	Ryumin-Light-Hojo-EUC-H
GothicBBB-Medium-Hojo-EUC-V	Ryumin-Light-Hojo-EUC-V

SJIS エンコーディングのフォント

GothicBBB-Medium-RKSJ-H	Ryumin-Light-RKSJ-H
GothicBBB-Medium-RKSJ-V	Ryumin-Light-RKSJ-V

83pv-RKSJ エンコーディングのフォント

```
GothicBBB-Medium-83pv-RKSJ-H Ryumin-Light-83pv-RKSJ-H  
GothicBBB-Medium-83pv-RKSJ-V Ryumin-Light-83pv-RKSJ-V
```

組み込まれたフォント名は、各フォント名に `New` を付けたフォント名になります。たとえば、`GothicBBB-Medium-EUC-H` にユーザー定義文字を追加したフォントのフォント名は、`NewGothicBBB-Medium-EUC-H` になります。

ユーザー定義文字の印刷方法

「Solaris 外字ツール」を使って登録したユーザー定義文字を印刷するには、`jpostprint(1)` を使って、いったん PostScript ファイルを生成してから、次のスクリプトで印刷します (`jpostprint` はデフォルトでは、`$HOME/.Xlocale/$LANG/fonts/UDC` ディレクトリに保存されているユーザー定義文字を使用します)。

```
sun% /usr/lib/lp/postscript/jpostprint <テキストファイル名> | lpr
```

環境変数 `DTUDCFONTPATH` で保存ディレクトリを設定した場合は、次のように指定します。

```
sun% /usr/lib/lp/postscript/jpostprint -u "$DTUDCFONTPATH/UDC%d.pfa" | lpr
```

詳細は、`jpostprint(1)` のマニュアルページを参照してください。

オプション

次のオプションを使用できます。

```
sun% /usr/dt/bin/sdtudctool -f fontfile
```


「Solaris 外字ツール」は起動時に指定したフォントファイルを読み込みます。現時点では、BDF/PCF 形式のフォントを読み込むことができます。

環境設定

環境変数

DTUDCFONTPATH

ユーザー定義文字を保存するディレクトリを指定します。指定しない場合、以下のディレクトリがユーザー定義文字の保存ディレクトリとして利用されます。

ローカルユーザーの場合

```
$HOME/.Xlocale/$LANG/fonts/UDC
```

スーパーユーザーの場合

```
$OPENWINHOME/lib/locale/$LANG/X11/fonts/UDC
```

リソース

以下のリソースをサポートしています。

utUDCPrefix

ユーザー定義文字を保存するビットマップフォントファイルの接頭辞を指定します。デフォルトは UDC です。

utUDCBDFSize

ユーザー定義文字のビットマップフォントのサイズを指定します。デフォルトは 12、14、16、20、24 です。

utRefXFont

参照一覧を開いた時の表示文字を XLFD 名で指定します。デフォルトは
-sun-gothic-medium-r-normal--16-140-75-75-c-140-jisx0208.1983-0
です。

utUDCCIDPrefix

ユーザー定義文字を組み込んだ際のフォントの名の接頭辞です。デフォルトは New
です。

utUDCCIDBase

ユーザー定義文字を組み込むフォント名からエンコード名以降を除いた文字です。
デフォルトは GothicBBB-Medium と Ryumin-Light です。

フォントの移行

この章では、すでに利用しているフォントの移行方法について説明します。

Solaris1.x SunView で使用していた evfont の再利用

JLE SunView システムで使用していた evfont フォントは、次の変換作業を順番に行うことにより、日本語 OpenWindows 3.6 上で使用可能になります。

1. evfont から **Adobe/MIT BDF 2.1** への変換
2. **Adobe/MIT BDF 2.1** から **PCF** フォントファイルへの変換
3. フォントエイリアスの追加
4. フォントセットの定義

evfont から **Adobe/MIT BDF 2.1** への変換

まず、evftobdf(1) コマンドを使用して、いったん evfont を Adobe/MIT BDF 2.1 フォントファイル (拡張子 .bdf) に変換します。

```
sun% evftobdf jpn.gotm.12
jpn.gotm.12->jpn.gotm.12.j201.bdf
jpn.gotm.12->jpn.gotm.12.j208.bdf
```

1つの evfont フォントファイルから2つの Adobe/MIT BDF 2.1 フォントファイルが出力されます。.j201.bdf の拡張子の付いたファイルは、JIS X 0201 の文字集合のフォントを表し、.j208.bdf の拡張子の付いたファイルは、JIS X 0208 の文字集合のフォントを表します。

Adobe/MIT BDF 2.1 から PCF フォントファイルへの変換

次に、OpenWindows 標準コマンドの `bdf2pcf` (1) を使用して、Adobe/MIT BDF 2.1 フォントファイルをサーバーがアクセス可能な PCF フォントファイルへ変換します。

```
sun% /usr/openwin/bin/bdf2pcf jpn.gotm.12.j208.bdf -o jpn.gotm.12.j208.pcf
sun% /usr/openwin/bin/bdf2pcf jpn.gotm.12.j201.bdf -o jpn.gotm.12.j201.pcf
```

フォントエイリアスの追加

フォントファイルの変換が終了したあと、X11 ウィンドウシステムからアクセスするために、変換したフォントに対する XLFD フォント名のエイリアスを追加します。

フォントエイリアスの追加は、`mkfontdir` (1) コマンドを実行することにより、自動的に行うことができますが、そのためには前述の .bdf ファイルの FONT フィールドの XLFD の記述が適切である必要があります。XLFD 記述規約や設定値の詳細は、『X プロトコルリファレンス・マニュアル 0 巻』を参照してください。

```
sun% mkfontdir fontdir1
```

1. `fontdir` には、フォントが置かれているディレクトリを指定します。

なお、フォントディレクトリに `font.alias` ファイルを置くことにより、フォントのエイリアスとフォント名パターンの定義が可能となります。

```
gotm-12 -sun-gothic-medium-r-normal--12-120-75-75-c-120-japanese-0
```

フォントセットの定義

変換したフォントを、さらに日本語フォントセットとして DeskSet などの日本語 OpenWindows のアプリケーションから使用するには、日本語 OpenWindows のフォントセット定義ファイルに、エイリアスとして追加した XLFD フォント名の登録が必要です。フォントセット定義方法の詳細は、『XView Developer's Notes』を参照してください。

ここでは、例とし

て、`/usr/openwin/lib/locale/ja/OW_FONT_SETS/OpenWindows.fs` に以下の行を追加します。なお、この作業終了後、OpenWindows を立ち上げ直す必要があります。

```
! sample definition
!
gotm-12:alias,\
  -sun-gothic-medium-r-normal--12-120-75-75-c-120-japanese-0

  -sun-gothic-medium-r-normal--12-120-75-75-c-120-japanese-0:definition,\
  -sun-gothic-medium-r-normal--12-120-75-75-c-60-jisx0201.1976-0,\
  -sun-gothic-medium-r-normal--12-120-75-75-c-120-jisx0208.1983-0
```

evfont から変換したフォントファイルは、上記の例の場合、次の方法で確認できます。

```
sun% xfd -fn gotm-12
```

Solaris 2.5.1 以前のシステムで作成したユーザー定義文字の移行

Solaris 2.5.1 以前の環境ですでにユーザー定義文字を登録している場合は、次の方法でその文字を移行し、既存のユーザー定義文字を再利用できます。

ここでは、ユーザー定義文字を含む既存のフォントファイルから `$HOME/.Xlocale/$LANG/fonts/UDC` ディレクトリにユーザー定義文字を抜き出し、Solaris 7 でサポートする環境へ移行する方法について説明します。

なお、環境変数 DTUDCFONTPATH が設定されている場合は、そのディレクトリを利用してください。

フォントエディタで作成したビットマップフォントを移行する場合

1. sdtudc_extract を使用して、登録済みのユーザー定義文字を別フォントファイルに抜き出します。

sdtudc_extract は /usr/dt/config/locale/sdtudc_map にある変換テーブルを参照し、ユーザー定義文字のコードポイントを Solaris 7 のユーザー定義文字の領域内に移動しながら、別フォントファイルに抜き出します。

sdtudc_map の形式は「変換前の領域」と「変換後の領域の先頭」で表現され、デフォルトでは次のように記述されます。

a9a1,a9ff	f5a1
aaa1,aa1f	f6a1
aba1,abff	f7a1
aca1,acff	f8a1
ada1,adff	f9a1
aea1,aeff	faa1
afa1,afff	fb1

上記の 1 行目では、指定したフォントファイル中の 0xa9a1 から 0xa9ff に登録されているユーザー定義文字を 0xf5a1 から始まるコードポイントに順番に割り付けながら抜き出すことを表しています。

```
sun% sdtudc_extract gotml4.pcf > UDC14.bdf
```

14 ドットのビットマップフォントの場合は、作成するフォントファイル名を UDC14.pcf にしてください。

2. 抜き出したフォントファイルをユーザー定義文字の保存ディレクトリに移動してから、「Solaris 外字ツール」を起動し、新たにユーザー定義文字を登録します。

```
sun% bdf2pcf -o UDC14.pcf UDC14.bdf
sun% mkdir -p $HOME/.Xlocale/$LANG/fonts/UDC/Bitmaps
sun% mv UDC14.pcf $HOME/.Xlocale/$LANG/fonts/UDC/Bitmaps
sun% /usr/dt/bin/sdtudctool
```

また、ユーザー定義文字を使用して作成したテキストファイルがある場合は、次のように `sdtudc_convert` を使ってユーザー定義文字のコードポイントを移動します。

```
sun% sdtudc_convert <テキストファイル> > <新しいテキストファイル>
```

`sdtudc_convert` のソースファイルは、ディレクトリ `$OPENWINHOME/share/src/locale/ja/fonts/sdtudc_convert.c` にあります。

詳細は、`sdtudc_convert(1)` のマニュアルページを参照してください。

フォントマネージャで作成したフォントを移行する場合

1. `sdtudc_extract_ps` を使って、登録済みのユーザー定義文字を別フォントファイルに抜き出します。

たとえば、フォントマネージャを使って作成した `font.ps` というフォントファイルからユーザー定義文字を移行するには、次のコマンドを実行します。

```
sun% sdtudc_extract_ps font.ps
UDC1.pfa
UDC2.pfa
UDC3.pfa
:
UDC10.pfa
```

2. 抜き出したフォントファイルをユーザー定義文字の保存ディレクトリに移動してから、`sdtudctool` を起動し、新たなユーザー定義文字を登録します。

```
sun% mkdir -p $HOME/.Xlocale/$LANG/fonts/UDC/Type1
sun% mv UDC*.pfa $HOME/.Xlocale/$LANG/fonts/UDC/Type1
sun% /usr/dt/bin/sdtudctool
```

フォントエディタで作成したビットマップフォントと フォントマネージャで作成したフォントをマージする場合

1. フォントエディタで作成したビットマップフォントファイルとフォントマネージャで作成したフォントファイルからユーザー定義文字を抜き出します。

```
sun% sdtudc_extract gotml4.pcf > UDC14.bdf
sun% sdtudc_extract_ps font.ps
UDC1.pfa
UDC2.pfa
UDC3.pfa
:
UDC10.pfa
```

2. 「Solaris 外字ツール」を起動したときに参照するディレクトリに移動します。
フォントエディタで作成したフォントファイルを優先する場合

```
sun% mkdir -p $HOME/.Xlocale/$LANG/fonts/UDC/Bitmaps
sun% mv UDC14.bdf $HOME/.Xlocale/$LANG/fonts/UDC/Bitmaps
```

フォントマネージャで作成したフォントを優先する場合

```
sun% mkdir -p $HOME/.Xlocale/$LANG/fonts/UDC/Type1
sun% mv UDC*.pfa $HOME/.Xlocale/$LANG/fonts/UDC/Type1
```

3. 「Solaris 外字ツール」を起動します。
4. 参照表を開き、もう一方のフォントファイルを読み込みます。

```
sun% /usr/dt/bin/sdtudctool
```
5. 参照表から、必要なユーザー定義文字を一覧表にドラッグ&ドロップします。
6. 「ファイル」→「保存」を選択し、ユーザー定義文字フォントファイルを作成します。

7. 「ファイル」 → 「ユーザー定義文字」を選択し、ユーザー定義文字を読み込んでから新たに登録を開始します。

文字コード変換規則

この付録では、文字コード変換基規則について説明します。また、IBM 拡張文字の変換表も記載します。

新しい文字コード変換規則

Solaris 2.6 から、『TOG 日本ベンダ協議会推奨 日本語 EUC・シフト JIS コード変換仕様』に基づく新しいコード変換規則が導入されました。

この変換規則に従うものは次の通りです。

- `iconv(1)` コマンド
- `iconv(3)` インタフェース
- `jconv(7M)` および `jconvrs(7M)` STREAMS モジュール

次に変換規則を図示します。

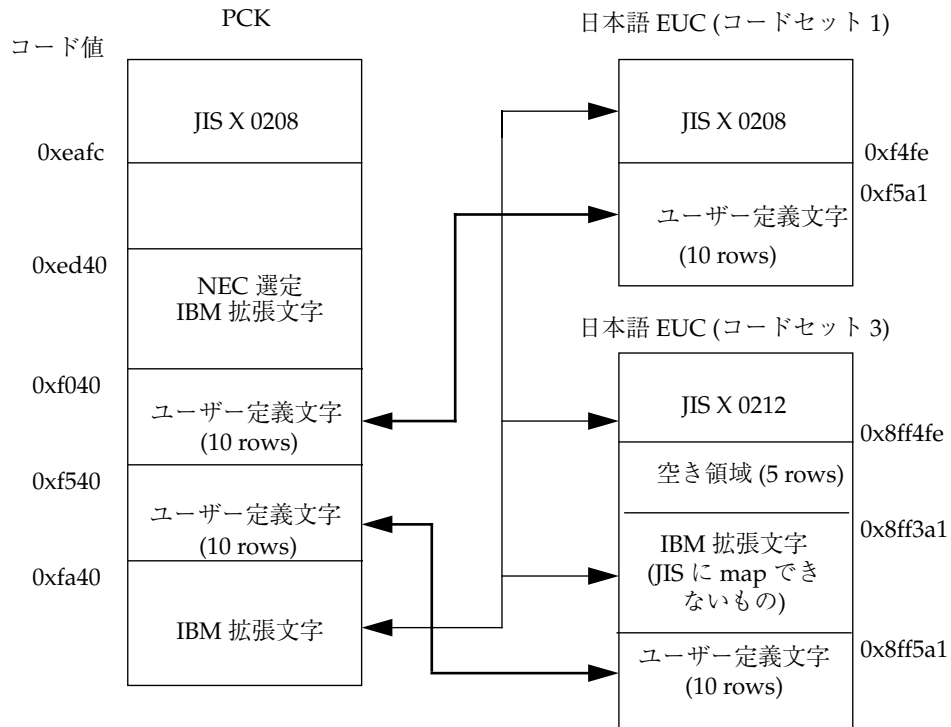


図 A-1 文字コード変換規則

図 A-1 に関する注意事項を次に示します。

PCK と日本語 EUC の間では、共通自由領域にユーザー定義文字および IBM 拡張文字を次のように割り当てるものとします。

	PCK	日本語 EUC
ユーザー定義文字	0xf040 - 0xf4fc	0xf5a1 - 0xfefe
1880 文字	0xf540 - 0xf9fc	0x8ff5a1 - 0x8ffefe
IBM 拡張文字	0xfa40 - 0xf4fc	JISX0208
388 文字		JISX0212
		0x8ff3a1 - 0x8ff4fe

- ユーザー定義文字は区番号、点番号ともに番号の小さいほうから順にコードを割り当てるとします。
- IBM 拡張文字は JIS X 0208、JIS X 0212 に対応する文字がある場合はその文字に変換し、それ以外の文字は 84 区 94 点 からコードの値が小さいほうへ順に変換します。
- この変換では、コードセット 3 の 83 区 1 点 ~ 82 点は予約領域です。
- JIS X 0212 は、日本語 EUC から PCK への変換で PCK に対応する文字がない場合は、ある特定の文字 (以降「置換文字」とします) に変換します。
- NEC 選定 IBM 拡張文字は、PCK から日本語 EUC への変換では、PCK で一度 IBM 拡張文字に変換してから日本語 EUC に変換します。
- 日本語 EUC から PCK への変換で IBM 拡張文字から NEC 選定 IBM 拡張文字へ変換されません。
- 変換元の文字コードとしては有効ですが、変換先に対応する文字コードが存在しない場合は、「置換文字」に変換します。
- 「置換文字」として「=」を使用します。
- IBM 拡張文字の具体的な対応は、表 A-1を参照してください。

IBM 拡張文字の変換表

日本語 EUC と PCK との間で変換される IBM 拡張文字のコードポイントは、次の表のように対応付けられます。

注 - これにより、今回のリリースで変換が保証されている IBM 拡張文字とその内訳は、下記の通り計 388 文字となります。

JIS X 0208 領域中の文字 : 2 文字

JIS X 0212 領域中の文字 : 386 文字 (うち 83 ~ 84 区 へ 106 文字)

表 A-1 IBM 拡張文字の変換表

PCK	日本語 EUC	PCK	日本語 EUC	PCK	日本語 EUC	PCK	日本語 EUC
fa40	8ff3f3	fa41	8ff3f4	fa42	8ff3f5	fa43	8ff3f6
fa44	8ff3f7	fa45	8ff3f8	fa46	8ff3f9	fa47	8ff3fa
fa48	8ff3fb	fa49	8ff3fc	fa4a	8ff3fd	fa4b	8ff3fe
fa4c	8ff4a1	fa4d	8ff4a2	fa4e	8ff4a3	fa4f	8ff4a4
fa50	8ff4a5	fa51	8ff4a6	fa52	8ff4a7	fa53	8ff4a8
fa54	a2cc	fa55	8fa2c3	fa56	8ff4a9	fa57	8ff4aa
fa58	8ff4ab	fa59	8ff4ac	fa5a	8ff4ad	fa5b	a2e8
fa5c	8fd4e3	fa5d	8fdcdf	fa5e	8fe4e9	fa5f	8fe3f8
fa60	8fd9a1	fa61	8fb1bb	fa62	8ff4ae	fa63	8fc2ad
fa64	8fc3fc	fa65	8fe4d0	fa66	8fc2bf	fa67	8fbcf4
fa68	8fb0a9	fa69	8fb0c8	fa6a	8ff4af	fa6b	8fb0d2
fa6c	8fb0d4	fa6d	8fb0e3	fa6e	8fb0ee	fa6f	8fb1a7
fa70	8fb1a3	fa71	8fb1ac	fa72	8fb1a9	fa73	8fb1be
fa74	8fb1df	fa75	8fb1d8	fa76	8fb1c8	fa77	8fb1d7
fa78	8fb1e3	fa79	8fb1f4	fa7a	8fb1e1	fa7b	8fb2a3
fa7c	8ff4b0	fa7d	8fb2bb	fa7e	8fb2e6	fa80	8fb2ed
fa81	8fb2f5	fa82	8fb2fc	fa83	8ff4b1	fa84	8fb3b5
fa85	8fb3d8	fa86	8fb3db	fa87	8fb3e5	fa88	8fb3ee
fa89	8fb3fb	fa8a	8ff4b2	fa8b	8ff4b3	fa8c	8fb4c0
fa8d	8fb4c7	fa8e	8fb4d0	fa8f	8fb4de	fa90	8ff4b4
fa91	8fb5aa	fa92	8ff4b5	fa93	8fb5af	fa94	8fb5c4
fa95	8fb5e8	fa96	8ff4b6	fa97	8fb7c2	fa98	8fb7e4
fa99	8fb7e8	fa9a	8fb7e7	fa9b	8ff4b7	fa9c	8ff4b8

表 A-1 IBM 拡張文字の変換表 続く

PCK	日本語 EUC	PCK	日本語 EUC	PCK	日本語 EUC	PCK	日本語 EUC
fa9d	8ff4b9	fa9e	8fb8ce	fa9f	8fb8e1	faa0	8fb8f5
faa1	8fb8f7	faa2	8fb8f8	faa3	8fb8fc	faa4	8fb9af
faa5	8fb9b7	faa6	8fbabe	faa7	8fbadb	faa8	8fcdaa
faa9	8fbae1	faaa	8ff4ba	faab	8fbaeb	faac	8fbbb3
faad	8fbbb8	faae	8ff4bb	faaf	8fbbca	fab0	8ff4bc
fab1	8ff4bd	fab2	8fbbd0	fab3	8fbbde	fab4	8fbbf4
fab5	8fbbf5	fab6	8fbbf9	fab7	8fbce4	fab8	8fbced
fab9	8fbcfe	faba	8ff4be	fabb	8fbdc2	fabc	8fbde7
fabd	8ff4bf	fabe	8fbd0f	fabf	8fbef0	fac0	8fbeac
fac1	8ff4c0	fac2	8fbef3	fac3	8fbefd	fac4	8fbecd
fac5	8fbec9	fac6	8fbee4	fac7	8fbfa8	fac8	8fbfc9
fac9	8fc0c4	faca	8fc0e4	facb	8fc0f4	facc	8fc1a6
facd	8ff4c1	face	8fc1f5	facf	8fc1fc	fad0	8ff4c2
fad1	8fc1f8	fad2	8fc2ab	fad3	8fc2a1	fad4	8fc2a5
fad5	8ff4c3	fad6	8fc2b8	fad7	8fc2ba	fad8	8ff4c4
fad9	8fc2c4	fada	8fc2d2	fadb	8fc2d7	fadc	8fc2db
fadd	8fc2de	fade	8fc2ed	fadf	8fc2f0	fae0	8ff4c5
fae1	8fc3a1	fae2	8fc3b5	fae3	8fc3c9	fae4	8fc3b9
fae5	8ff4c6	fae6	8fc3d8	fae7	8fc3fe	fae8	8ff4c7
fae9	8fc4cc	faea	8ff4c8	faeb	8fc4d9	faec	8fc4ea
faed	8fc4fd	faee	8ff4c9	faef	8fc5a7	faf0	8fc5b5
faf1	8fc5b6	faf2	8ff4ca	faf3	8fc5d5	faf4	8fc6b8

表 A-1 IBM 拡張文字の変換表 続く

PCK	日本語 EUC	PCK	日本語 EUC	PCK	日本語 EUC	PCK	日本語 EUC
faf5	8fc6d7	faf6	8fc6e0	faf7	8fc6ea	faf8	8fc6e3
faf9	8fc7a1	fafa	8fc7ab	fafb	8fc7c7	fafc	8fc7c3
fb40	8fc7cb	fb41	8fc7cf	fb42	8fc7d9	fb43	8ff4cb
fb44	8ff4cc	fb45	8fc7e6	fb46	8fc7ee	fb47	8fc7fc
fb48	8fc7eb	fb49	8fc7f0	fb4a	8fc8b1	fb4b	8fc8e5
fb4c	8fc8f8	fb4d	8fc9a6	fb4e	8fc9ab	fb4f	8fc9ad
fb50	8ff4cd	fb51	8fc9ca	fb52	8fc9d3	fb53	8fc9e9
fb54	8fc9e3	fb55	8fc9fc	fb56	8fc9f4	fb57	8fc9f5
fb58	8ff4ce	fb59	8fcab3	fb5a	8fcabd	fb5b	8fcaef
fb5c	8fcaf1	fb5d	8fcbae	fb5e	8ff4cf	fb5f	8fcba
fb60	8fcbe6	fb61	8fcbea	fb62	8fcbf0	fb63	8fcbf4
fb64	8fcbee	fb65	8fccaa5	fb66	8fcbf9	fb67	8fccab
fb68	8fccae	fb69	8fccad	fb6a	8fccb2	fb6b	8fccc2
fb6c	8fccd0	fb6d	8fccd9	fb6e	8ff4d0	fb6f	8fccdbb
fb70	8ff4d1	fb71	8fcebb	fb72	8ff4d2	fb73	8fceba
fb74	8fcec3	fb75	8ff4d3	fb76	8fcef2	fb77	8fb3dd
fb78	8fcfd5	fb79	8fcfe2	fb7a	8fcfe9	fb7b	8fcfed
fb7c	8ff4d4	fb7d	8ff4d5	fb7e	8ff4d6	fb80	8ff4d7
fb81	8fd0e5	fb82	8ff4d8	fb83	8fd0e9	fb84	8fd1e8
fb85	8ff4d9	fb86	8ff4da	fb87	8fd1ec	fb88	8fd2bb
fb89	8ff4db	fb8a	8fd3e1	fb8b	8fd3e8	fb8c	8fd4a7
fb8d	8ff4dc	fb8e	8ff4dd	fb8f	8fd4d4	fb90	8fd4f2

表 A-1 IBM 拡張文字の変換表 続く

PCK	日本語 EUC	PCK	日本語 EUC	PCK	日本語 EUC	PCK	日本語 EUC
fb91	8fd5ae	fb92	8ff4de	fb93	8fd7de	fb94	8ff4df
fb95	8fd8a2	fb96	8fd8b7	fb97	8fd8c1	fb98	8fd8d1
fb99	8fd8f4	fb9a	8fd9c6	fb9b	8fd9c8	fb9c	8fd9d1
fb9d	8ff4e0	fb9e	8ff4e1	fb9f	8ff4e2	fba0	8ff4e3
fba1	8ff4e4	fba2	8fdcd3	fba3	8fddc8	fba4	8fddd4
fba5	8fddea	fba6	8fddfa	fba7	8fdea4	fba8	8fdeb0
fba9	8ff4e5	fbaa	8fdeb5	fbab	8fdecb	fbac	8ff4e6
fbad	8fdfb9	fbae	8ff4e7	fbaf	8fdfc3	fbb0	8ff4e8
fbb1	8ff4e9	fbb2	8fe0d9	fbb3	8ff4ea	fbb4	8ff4eb
fbb5	8fe1e2	fbb6	8ff4ec	fbb7	8ff4ed	fbb8	8ff4ee
fbb9	8fe2c7	fbba	8fe3a8	fbbb	8fe3a6	fbbc	8fe3a9
fbbd	8fe3af	fbbe	8fe3b0	fbbf	8fe3aa	fbc0	8fe3ab
fbc1	8fe3bc	fbc2	8fe3c1	fbc3	8fe3bf	fbc4	8fe3d5
fbc5	8fe3d8	fbc6	8fe3d6	fbc7	8fe3df	fbc8	8fe3e3
fbc9	8fe3e1	fbca	8fe3d4	fbcb	8fe3e9	fbcc	8fe4a6
fbcd	8fe3f1	fbce	8fe3f2	fbcf	8fe4cb	fdb0	8fe4c1
fdb1	8fe4c3	fdb2	8fe4be	fdb3	8ff4ef	fdb4	8fe4c0
fdb5	8fe4c7	fdb6	8fe4bf	fdb7	8fe4e0	fdb8	8fe4de
fdb9	8fe4d1	fbda	8ff4f0	fbdb	8fe4dc	fbdc	8fe4d2
fbdd	8fe4db	fbde	8fe4d4	fbdf	8fe4fa	fbe0	8fe4ef
fbe1	8fe5b3	fbe2	8fe5bf	fbe3	8fe5c9	fbe4	8fe5d0
fbe5	8fe5e2	fbe6	8fe5ea	fbe7	8fe5eb	fbe8	8ff4f1

表 A-1 IBM 拡張文字の変換表 続く

PCK	日本語 EUC	PCK	日本語 EUC	PCK	日本語 EUC	PCK	日本語 EUC
fbe9	8ff4f2	fbea	8ff4f3	fbeb	8fe6e8	fbec	8fe6ef
fbed	8fe7ac	fbee	8ff4f4	fbef	8fe7ae	fbf0	8ff4f5
fbf1	8fe7b1	fbf2	8ff4f6	fbf3	8fe7b2	fbf4	8fe8b1
fbf5	8fe8b6	fbf6	8ff4f7	fbf7	8ff4f8	fbf8	8fe8dd
fbf9	8ff4f9	fbfa	8ff4fa	fbfb	8fe9d1	fbfc	8ff4fb
fc40	8fe9ed	fc41	8feacd	fc42	8ff4fc	fc43	8feadb
fc44	8feae6	fc45	8feaea	fc46	8feba5	fc47	8febfb
fc48	8febfa	fc49	8ff4fd	fc4a	8fecd6	fc4b	8ff4fe

索引

A

AccessX, 103
Admintool, 「システム管理ツール」, を参照,
alpha, 15
AnswerBook, 107
ATOK8
日本語入力の切り替え, 97

B

biff コマンド, 81
blank, 15

C

cntrl, 14
Common Desktop Environment, 93
comsat, 81
crt 変数, 74
cs00
日本語入力の切り替え, 97
cs00 ユーザー辞書ツール, 107, 113
.cshrc, 29, 30, 32

D

DeskSet, 111
digit, 14
dtlogin コマンド, 93
DTUDCFONTPATH, 161
dumb プリンタ, 37

E

env コマンド, 69
EUC (Extended UNIX Code), 9
euctosj コマンド, 6
EUC コードセット 3, 102, 126
evfont, 163

F

folderconv 変数, 75, 90
fonts ディレクトリ, 101

G

graph, 16
GUI 環境, 93
起動, 93
ロケール, 94

H

HG ゴシック体 B, 99, 100
HG 明朝体 L, 99, 100
htt コマンド, 98

I

IBM 拡張文字, 101
コードポイント, 173
変換表, 173
iconv(1), 171
iconv(3), 171
iconv コマンド, 6, 83

コード変換, 84
in.comsat コマンド, 36
 コード変換, 90
indentfix 変数, 80
install_comsat コマンド, 6

J

jalpha, 19
japanese ロケール, 94, 125
ja_JP.PCK ロケール, 10, 94
ja_JP.UTF-8 ロケール, 102
ja ロケール, 9, 25, 94
jconvrs(7M), 171
jconvrs(7M), 171
jdigit, 17
JFP (Japanese Feature Package), 3
 パッケージ, 4
jgen, 21
jgreek, 20
jhira, 17
jisx0201r, 17
jisx0208, 18
JIS X 0208-1990 文字集合, 100
jisx0212, 18
JIS X 0212
 印刷, 51
JIS X 0212-1990 補助漢字, 101, 102, 126
jkanji, 16
jkata, 17
JLE
 コード変換フィルタ, 86
 互換性, 125
jparen, 17
jpostprint
 オプション, 39
jprconv
 オプション, 38
jpunct, 21
jrussian, 20
jsci, 21
jspecial, 20
junit, 20

K

Kanji.font, 101

Kodak Color Management System
(KCMS), 103

L

LANG, 25, 119
LC_COLLATE カテゴリ, 21
LC_CTYPE カテゴリ, 12
LC_TIME カテゴリ, 22
LED, 127
line, 17
locale, 3
localization, 3
.login, 25, 31
lower, 13
LP 印刷サービス, 7, 36
lp コマンド
 オプション, 37
ls コマンド, 28, 30

M

.mailrc, 74
mailx, 「メールツール」, を参照,
MANPATH, 30
man コマンド, 69
MB_CONSERVATIVE_EDIT, 65
Motif, 103
Motif ツールキット
 日本語メッセージ, 113
MS 漢字コード, 10

N

NEC 選定ユーザー定義文字, 101
NEC 特殊文字, 101
NLSPATH, 113
nroff コマンド, 69

O

OLIT, 116, 117
OLIT アプリケーション, 116
OPENLOOK, 112

P

PAGER 変数, 74

PATH パラメタ, 29
path 変数, 32
PC 漢字コード, 5, 10, 12
pkginfo コマンド, 4
PostScript, 39
print, 15
printers.conf ファイル, 50
.profile, 26, 31, 32
punct, 14

R

record 変数, 75
regex コマンド, 13

S

sdtudctool, 「Solaris 外字ツール」, を参照,
sdtudicm, 107
setterm コマンド, 7, 26, 28, 89
set コマンド, 29
sjtojis コマンド, 6
Solaris, 3
Solaris CDE
 アプリケーション, 105
 概要, 105
 カスタマイズ, 108
 機能, 105
 推奨環境, 110
Solaris 外字ツール, 106
 アウトライン, 144
 一覧表, 134, 147
 オプション, 160
 環境設定, 161
 起動方法, 135
 機能, 101, 131, 132, 136
 参照表, 135, 149
 ドラッグ&ドロップ, 151
 ビットマップ, 145
 描画ツール, 137
 フォントパスの設定, 142
 フォントパスの保存, 143
 ヘルプ, 147
 文字エディタ, 133, 136
 リソース, 161
source コマンド, 29
space, 14
STREAMS モジュール, 7, 88, 171

strftime(3C), 22
SunOS, 3
SunOS コマンド, 5
 国際化, 55
 日本語環境, 83
SunOS メッセージ, 5
SunView
 フォントの再利用, 163

T

talk コマンド, 6
ToolTalk, 104
troff コマンド, 69
tty, 26, 33

U

udc, 18
udicmtool, 113
upper, 13
UTF-8, 11
utRefXFont, 162
utUDCBDFSize, 161
utUDCCIDBase, 162
utUDCCIDPrefix, 162
utUDCPrefix, 161

V

vdc, 19
vi エディタ, 55
 英語環境, 65
 大文字・小文字変換, 59, 60
 改行, 63
 起動, 56
 行の分割, 63
 行の連結, 63
 行末処理, 57
 削除, 59
 終了, 67
 処理, 57
 挿入, 57
 単語単位の日本語処理, 60
 置換, 59
 ファイルの保存, 67
 マージン, 63
 文字列の検索, 65

文字列の置換, 65
連結, 63

W

wctype(3C), 13
which コマンド, 29
windex, 70
Wnn6
日本語入力の切り替え, 97

X

X11R6 XIM プロトコル, 98
xdigit, 15
XGL 漢字ストロークフォント, 101
XGL グラフィックスライブラリ, 103
XIL イメージ処理ライブラリ, 103
xlsfonts コマンド, 100
XPG4, 29
xserverconfig, 104
XView, 117
XView アプリケーション, 115

い

インストールツール, 103

お

オンラインマニュアル, 69, 70
apropos コマンド, 71
whatis コマンド, 71

か

拡張 UNIX コード, 9
仮想 LED, 104, 127
オプション, 128
起動, 127
各国語対応 (localization), 3
環境変数, 30
LANG, 25, 119
MANPATH, 30
MB_CONSERVATIVE_EDIT, 65
ロケール, 25

き

キーテーブル, 54
キーボード, 53
キーボードのサポート, 8

く

クラス
文字分類, 12

け

言語, 94
検索パス
オンラインマニュアル, 30
コマンド, 28, 32

こ

コードセット 3, 102
コード変換, 6, 83
国際化 (internationalization), 4
国際化拡張コマンド, 6
ゴシック体, 100
ゴシック体ボールド, 100
午前/午後, 24
コマンド, 5

し

時刻, 24
システム管理ツール, 103
プリンタの追加, 41
システムロケール, 35
シフト JIS コード, 10
初期化ファイル
.cshrc, 27, 29 - 32
.login, 25, 27, 31
.profile, 26, 27, 29, 31, 32

す

スタイルマネージャ, 100, 108

せ

セッション, 94

た

代替年表示, 24
単語処理, 22
端末, 26
 コード変換, 7
端末ドライバ, 26

つ

月, 22

て

電源管理, 104

と

動作環境, 123

に

日本語 DeskSet
 プログラム, 111
日本語 EUC, 5, 9, 12, 25
日本語 Magnify Help, 114
日本語 OpenWindows, 111
 移行, 120, 121
 概要, 111
 カスタマイズ, 123
 起動, 119
 機能, 111
 基本ロケール, 125
 時刻形式, 126
 初期設定, 119
 数値, 126
 入力ロケール, 125
 表示ロケール, 125
 プロパティ, 123
 ユーザー定義文字, 132
 ロケールの切り替え, 124
日本語 OpenWindows 2.0.1
 移行, 120
日本語 OpenWindows 3.2
 移行, 121
日本語 PostScript プリンタ, 39
 印刷, 52
 印刷フィルタ, 46
日本語 TrueType フォント, 99

日本語 dumb プリンタ
 印刷, 51
 印刷フィルタ, 46
日本語 オンラインヘルプ
 Solaris CDE, 107
 日本語 OpenWindows, 114
日本語 オンラインマニュアル, 8, 115
日本語 環境, 25
日本語 キーボード, 8, 53
日本語 対応, 3
日本語 タイプ 4、5 キーボード, 54
日本語 入力, 7
 ATOK8, 7
 Solaris CDE, 108
 Wnn6, 7
 cs00, 7, 113
 日本語入力の切り替え, 97
日本語 入力サーバー, 98
日本語 ビットマップフォント, 100
日本語 フォント, 99
 OLIT アプリケーション, 116
 Solaris CDE, 108
 XView アプリケーション, 115
 ディレクトリ, 101
 日本語 OpenWindows, 115
日本語 プリンタ, 7, 36
日本語 ヘルプハンドブック
 日本語 OpenWindows, 114
日本語 文字コード変換, 88
日本語 モジュール, 6
日本語 ロケール, 5, 9, 12
 ja_JP.PCK ロケール, 5, 12
 japanese ロケール, 5, 12
 ja ロケール, 5, 12

は

パッケージ, 4
パラメタ
 パス, 29

ひ

日付, 24
表示形式, 22

ふ

フォントエディタ
フォントの再利用, 166
フォントマネージャ
フォントの再利用, 167
プリンタ, 7, 36
Solaris CDE, 109
コントロールコード, 45
タイプ, 45
追加, 41
リモートプリンタ, 48
ローカルプリンタ, 41

へ

平成明朝体 W3H, 99, 100
ヘルプ
Solaris CDE, 107
日本語 OpenWindows, 114
ベンダー定義文字, 10, 11, 16, 18, 19, 51

ほ

ボリューム管理ユーザーインタフェース, 113

ま

マイクロソフト標準キャラクタセット, 10
マウスレスオペレーション, 121
マニュアルページ
catman, 70
英文, 69
参照, 69

み

明朝体, 100
明朝体ボード, 100

め

メールツール
Subject, 72
mbox, 78
ncopy コマンド, 77
nsave コマンド, 77
起動, 72
コード変換, 90

終了, 81
送信, 74
送信メッセージの保存, 75
挿入, 80
中止, 75
通知, 81
到着メッセージの保存, 77
編集, 78
返答, 79
メッセージの作成, 72
メッセージの挿入, 80
読み取り, 76

も

文字クラス名, 13
文字コード, 86
文字コード変換規則, 172
注意事項, 172
文字の照合順序, 21
文字分類, 12
文字分類クラス, 13

ゆ

ユーザー定義文字, 16, 18, 51
DPS 上での利用方法, 159
印刷方法, 160
ディレクトリ, 161
登録, 153
登録可能領域, 131
フォント, 101
利用方法, 156

よ

曜日, 23

ろ

ログイン, 94
ログインサーバー, 93
ロケール, 3, 12, 25, 27, 35
ロックキー, 127

わ

ワークスペースメニュー

Solaris CDE, 97
日本語 OpenWindows, 98