



Java Desktop System Release 2 シ ステム管理

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

Part No: 817-7792-10
2004 年 7 月

Copyright 2004 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

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。サン・マイクロシステムズ株式会社の書面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。

Federal Acquisitions: Commercial Software—Government Users Subject to Standard License Terms and Conditions.

本製品に含まれる HG-MinchoL、HG-MinchoL-Sun、HG-PMinchoL-Sun、HG-GothicB、HG-GothicB-Sun、および HG-PCGothicB-Sun は、株式会社リコーがリョービ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ースマスタをもとに作成されたものです。HeiseiMin-W3H は、株式会社リコーが財団法人日本規格協会からライセンス供与されたタイプフェースマスタをもとに作成されたものです。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、docs.sun.com、Sun のロゴ、AnswerBook、AnswerBook2 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。

サンのロゴマークおよび 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、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、オムロンソフトウェア株式会社で共同開発されたソフトウェアです。© Copyright OMRON Co., Ltd. 1995-2000. All Rights Reserved. © Copyright OMRON SOFTWARE Co., Ltd. 1995-2002 All Rights Reserved.

「ATOK」は、株式会社ジャストシステムの登録商標です。

「ATOK Server/ATOK12」は、株式会社ジャストシステムの著作物であり、「ATOK Server/ATOK12」にかかる著作権その他の権利は、株式会社ジャストシステムおよび各権利者に帰属します。

本製品に含まれる郵便番号辞書 (7 桁/5 桁) は郵政事業庁が公開したデータを元に制作された物です (一部データの加工を行なっています)。

本製品に含まれるフェイスマーク辞書は、株式会社ビレッジセンターの許諾のもと、同社が発行する『インターネット・パソコン通信フェイスマークガイド '98』に添付のものを使用しています。© 1997 ビレッジセンター

Unicode は、Unicode, Inc. の商標です。

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 DiSpinBox 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。(© 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、明示的であるかを問わない、なんらの保証も行われないものとします。

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、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。

原典: *Java Desktop System Release 2 Administration Guid*

Part No: 817-6968-10

Revision A



041001 @ 9495



目次

はじめに 7

1	GConf の使用方法	13
	GConf の概要	13
	GConf リポジトリ	14
	GConf 設定ソース	15
	GConf スキーマ	16
	GConf スキーマ定義ファイル	17
	GConf デーモン	18
	GConf コマンドラインツール	18
	設定値の設定	21
	一般的な環境の設定	22
	HTTP プロキシの設定	22
	ワークスペース数の設定	22
	キーボードショートカットの設定	23
	パネルとパネルオブジェクトの設定	23
	ルック&フィールの設定	27
	フォントの設定	27
	背景の設定	28
	スプラッシュイメージの設定	28
	デフォルト設定値の復元	29
2	メニューのカスタマイズ	31
	メニューの概要	31
	ファイル抽象レイヤー	32
	vfolder とメニュー	32

	デスクトップエントリファイル	34
	ディレクトリエントリファイル	36
	メニューの編集	36
	メニューの追加	37
	メニューへの項目の追加	38
	メニューの設定を編集する	38
	メニュー項目を編集する	38
	メニューから項目を削除する	39
	ユーザーが変更できないメニューの設定	39
3	テーマのインストール	41
	テーマの概要	41
	テーマのインデックスファイル	42
	新しいコントロールオプションをインストールする	43
	新しいウィンドウフレームのオプションをインストールする	43
	新しいアイコンオプションをインストールする	44
	テーマ用の新しいアイコンのインストール	45
	カスタムコントロールオプションを作成する	46
4	フォントの設定	47
	フォント設定の概要	47
	フォント置換	47
	すべてのユーザーにフォントを追加する	48
	個別ユーザーにフォントを追加する	48
5	MIME タイプ	51
	MIME タイプの概要	51
	ファイルの MIME タイプの検出	52
	ファイルタイプ判別パターン	52
	パターンマスク	53
	MIME タイプレジストリ	53
	MIME タイプのためのアプリケーションの登録	56
	GNOME デスクトップへのアプリケーションの追加	58
6	スクリーンセーバーの設定	59
	スクリーンセーバーの概要	59
	スクリーンセーバーの設定	59

	スクリーンセーバーディスプレイの変更	61
	Linux 上でスクリーンセーバーディスプレイを追加する	61
	Solaris プラットフォーム上でスクリーンセーバーディスプレイを追加する	61
	Linux 上でスクリーンセーバーディスプレイを無効化する	61
	Solaris プラットフォーム上でスクリーンセーバーディスプレイを無効化する	62
7	セッション管理	63
	セッションの概要	63
	セッションのデフォルトの設定	64
	Solaris プラットフォーム上でのログインスクリプトの使用	64
8	ヘルプシステム	67
	ヘルプシステムの概要	67
	OMF ファイル	67
	ScrollKeeper カタログ作成システム	68
9	パフォーマンスの改善	69
	パフォーマンスの改善の概要	69
	CPU 使用率の低減	69
	CPU リソースをあまり必要としないコントロールテーマオプションを使用する	70
	CPU リソースをあまり必要としないウィンドウフレームオプションを使用する	70
	メニュー上のアイコンをオフにして CPU 使用率を減らす	71
	スプラッシュ画面をオフにして CPU 使用率を減らす	72
	パネルアニメーションをオフにして CPU 使用率を減らす	72
	ファイルマネージャのパフォーマンスの改善	72
	省リソースモードを有効化して CPU 使用率を減らす	75
	Solaris プラットフォームで mediaLib ライブラリを使用する	75
	X ウィンドウシステムのネットワークトラフィックの低減	76
	ネットワークトラフィックの少ないウィンドウフレームテーマオプションを使用する	76
	メニュー上のアイコンをオフにしてネットワークトラフィックを減らす	76
	省リソースモードを有効化してネットワークトラフィックを減らす	77
10	Java Desktop System の機能の無効化	79
	Java Desktop System の機能の無効化の概要	79

スクリーンロックを無効にする	79
ログアウトを無効にする	80
強制終了操作を無効にする	80
コマンドライン操作を無効にする	81
パネル設定を無効にする	81
アプリケーションを制限する	82

A 隠しディレクトリ 85

B SMB プリンタ設定に関するセキュリティ上の注意事項 87

用語集 89

索引 93

はじめに

『Java Desktop System Release 2 システム管理』では、以下のリリースの Java™ Desktop System を実行するシステムの管理方法を説明します。

- x86 アーキテクチャ版 Java Desktop System Release 2 (Solaris™ オペレーティングシステム用)
- Linux 版 Java Desktop System Release 2

このマニュアルに記載される情報の大部分は、Java Desktop System のすべてのリリースで共通です。共通ではない情報については、プラットフォームが示されます。

対象読者

このマニュアルは、Java Desktop System Release 2 を実行する 1 つまたは複数のシステムの管理者を対象にしています。Java Desktop System Release 2 の使用方法については、『Java Desktop System Release 2 ユーザーズガイド』を参照してください。

このマニュアルをお読みになる前に

このマニュアルを読む前に、以下のトピックについて知っておく必要があります。

- UNIX® システム管理
- XML (Extensible Markup Language) ファイルの構造
- システム管理を行う上での XML ファイルの使用方法

内容の紹介

このマニュアルの構成は次のとおりです。

- 第 1 章は、GConf を使用したユーザー設定の管理方法について説明しています。
- 第 2 章は、メニューの実装およびメニューのカスタマイズ方法について説明しています。
- 第 3 章は、Java Desktop System で利用可能なテーマの種類、テーマのインストール方法、およびカスタムテーマの作成方法について説明しています。
- 第 4 章は、Java Desktop System でのフォントの設定方法について説明しています。
- 第 5 章は、アプリケーションでどのように MIME タイプが検出されるかを説明しています。また、MIME タイプの登録方法、および Java Desktop System へのアプリケーションの追加方法についても説明しています。
- 第 6 章は、スクリーンセーバーを設定する方法について説明しています。この章では、スクリーンセーバーに利用可能なディスプレイの変更方法についても説明します。
- 第 7 章は、セッション管理の概要と、セッションのデフォルトを設定する方法について説明しています。また、セッションとログインスクリプトに関する情報も提供します。
- 第 8 章は、Java Desktop System のヘルプシステムについて説明しています。
- 第 9 章は、Java Desktop System のパフォーマンスを改善する方法について説明しています。
- 第 10 章は、Java Desktop System の特定の機能を無効にする方法について説明しています。
- 付録 A は、Java Desktop System がユーザーのホームディレクトリに追加する隠しディレクトリについて説明しています。
- 付録 B は、Microsoft Windows ネットワーク共有のプリンタ構成に関するセキュリティ上の注意事項について説明しています。
- 用語集は、このマニュアルで使用される用語の定義を説明しています。

関連マニュアル

次に、このマニュアルに関連するマニュアルを示します。

- *Java Desktop System Release 2 Accessibility Guide*
- *Java Desktop System Release 2 インストールガイド*

- 『Java Desktop System Release 2 ご使用にあたって』
- Java Desktop System Release 2 問題の解決方法
- Java Desktop System Release 2 ユーザーズガイド

Sun Documentation Online へのアクセス

docs.sun.com™ では Sun が提供しているオンラインマニュアルを参照することができます。サイト内をブラウズしたり、マニュアル名や特定の題目をキーワードにして検索したりすることができます。URL は <http://docs.sun.com> です。

Sun 製品のマニュアルの注文

Sun Microsystems では、一部の製品につき印刷版のマニュアルもご提供しています。マニュアルのリストおよびご注文方法については <http://docs.sun.com> の “Buy printed documentation” をご覧ください。

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、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例を示します。	<code>.login</code> ファイルを編集します。 <code>ls -a</code> を使用してすべてのファイルを表示します。 <code>system%</code>

表 P-1 表記上の規則 (続き)

字体または記号	意味	例
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区別して示します。	system% su password:
<i>AaBbCc123</i>	変数を示します。実際に使用する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。
『 』	参照する書名を示します。	『コードマネージャ・ユーザーズガイド』を参照してください。
「 」	参照する章、節、ボタンやメニュー名、強調する単語を示します。	第 5 章「衝突の回避」を参照してください。 この操作ができるのは、「スーパーユーザー」だけです。
\	枠で囲まれたコード例で、テキストがページ行幅を超える場合に、継続を示します。	sun% grep `^#define \ XV_VERSION_STRING'

コード例は次のように表示されます。

■ C シェル

```
machine_name% command y|n [filename]
```

■ C シェルのスーパーユーザー

```
machine_name# command y|n [filename]
```

■ Bourne シェルおよび Korn シェル

```
$ command y|n [filename]
```

■ Bourne シェルおよび Korn シェルのスーパーユーザー

```
# command y|n [filename]
```

[] は省略可能な項目を示します。上記の例は、*filename* は省略してもよいことを示しています。

| は区切り文字 (セパレータ) です。この文字で分割されている引数のうち 1 つだけを指定します。

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 キーを押します)。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

注－*** 下記の規則に該当する場合は、該当する規則のみ表記する。どちらも該当しない場合は、このセクションごと削除。***

- このマニュアルでは、英語環境での画面イメージを使っています。このため、実際に日本語環境で表示される画面イメージとこのマニュアルで使っている画面イメージが異なる場合があります。本文中で画面イメージを説明する場合には、日本語のメニュー、ボタン名などの項目名と英語の項目名が、適宜併記されています。
- このマニュアルでは、「x86」という用語は、Intel 32 ビット系列のマイクロプロセッサチップ、および AMD が提供する互換マイクロプロセッサチップを意味します。

第 1 章

GConf の使用方法

この章では、GConf を使用してユーザーの設定を管理する方法について説明します。

GConf の概要

GConf を使用するとJava Desktop System ユーザー設定の管理が容易になります。システム管理者は、GConf を使用して次のことが可能になります。

- すべてのユーザーの特定の設定に必須値を設定する。システム管理者はこの方法で、ユーザーが特定の設定を更新できるかどうかを制御できます
- すべてのユーザーの特定の設定にデフォルト値を設定する
- 設定の定義ファイルで指定されている設定の推奨値を使用する
- 各設定に関するマニュアルを読む

設定値がローカルに、またはネットワークを介して変更されると、GConf はアプリケーションに設定値の変更を通知します。このため、設定を変更すると、その設定を使用するすべてのアプリケーションが直ちに更新されます。

GConf には、次のコンポーネントがあります。

- ユーザー設定のリポジトリ
- gconfd-2 デーモン
- gconftool-2 コマンドラインツール

GConf リポジトリ

GConf リポジトリ内の各設定は、キーと値のペアで構成されます。GConf 設定キーは、アプリケーション設定に対応するリポジトリ内の要素です。たとえば、`/apps/gnome-session/options/show_splash_screen` 設定キーは、セッション設定ツールの「ログイン時にスプラッシュ画面を表示する」オプションに対応します。Java Desktop System のユーザーインターフェースには、GConf リポジトリ内のすべての設定キーが含まれるわけではありません。たとえば、Panel 設定ツールには `/apps/panel/global/tooltips_enabled` キーに対応するオプションは含まれていません。

リポジトリは、単純な階層のファイルシステムのように構成されています。リポジトリには、次のものが含まれます。

- GConf リポジトリを使用するアプリケーションに対応するディレクトリ。たとえば、このファイルシステムには、`/apps/gnome-terminal` というディレクトリが含まれます。
- 設定のカテゴリに対応するサブディレクトリ。たとえば、このファイルシステムには、`/apps/gnome-terminal/global` というディレクトリが含まれます。
- ディレクトリ内の設定キーの一覧、およびそのキーに関する情報を含む特殊ファイル。たとえば、HTTP プロキシ設定に関連するキーについての情報を持つファイルは、ディレクトリ `/system/http_proxy` にあります。
- すべての設定キーを記述するファイルを含む `/schemas` ディレクトリ。

通常、設定キーは、次の単純なデータ型を持ちます。

- 文字列
- 整数
- 文字列のリスト
- 整数のリスト

リポジトリ内の設定キーの形式は、リポジトリの読み取りに使用されるバックエンドモジュールに依存します。リポジトリの読み取りに XML (Extensible Markup Language) バックエンドモジュールが使用される場合の `/desktop/gnome/interface/font_name` 設定キーの例を、以下に示します。

```
<entry name="font_name" mtime="1038323555" muser="user123" type="string">
<stringvalue>Sans 10</stringvalue></entry>
```

注 - このマニュアルでは、設定キーを参照するときに、キーのパス名をそのキーの名前に追加しています。たとえば、`/desktop/gnome/interface` サブディレクトリ内の `font_name` 設定キーは、`/desktop/gnome/interface/font_name` のように参照されます。

GConf 設定ソース

GConf リポジトリは、「設定ソース」と呼ばれる一連の保存場所を含んでいます。設定ソースは、「GConf パスファイル」に記述されます。GConf パスファイルの場所は、`/etc/gconf/gconf-version-number/path` です。各ユーザーは、パスファイルを持っています。パスファイルは、各設定ソースに対して次の情報を指定します。

- リポジトリの読み取りに使用するバックエンドモジュール
- リポジトリのアクセス権
- リポジトリの場所

GConf パスファイルには、`include` 命令も含まれています。デフォルトでは、GConf パスファイルの内容は以下のようになります。

```
xml:readonly:/etc/gconf/gconf.xml.mandatory
include /etc/gconf/2/local-mandatory.path
apoc:readonly:mandatory@
include "$(HOME)/.gconf.path"
xml:readwrite:$(HOME)/.gconf
apoc:readonly:@
include /etc/gconf/2/local-defaults.path
xml:readonly:/etc/gconf/gconf.xml.defaults
```

GConf が設定値を検索する場合、GConf は、パスファイル内で指定された順に設定ソースを読みます。以下の表は、パスファイル内の設定ソースです。

設定ソース	説明
必須 (<code>xml:readonly:/etc/gconf/gconf.xml.mandatory</code>)	この設定ソースのアクセス権は、読み取り専用設定されます。ユーザーは、このソースの値を上書きすることができません。したがって、そのソースの設定値は必須です
ユーザー	この設定ソースは、ユーザーのホームディレクトリ内の <code>.gconf</code> ディレクトリに格納されます。ユーザーが環境を設定すると、新しい設定情報がこの場所に追加されます ユーザー設定ソースは、GConf エディタで変更できます。
デフォルト値	この設定ソースには、デフォルトの設定が含まれています

パスファイル内の設定ソースの順序では、必須の設定がユーザーの設定よりも優先されます。また、デフォルト設定よりもユーザーの設定が優先されます。つまり、GConf は、次の優先順位で設定を適用します。

1. 必須の設定
2. ユーザー指定の設定
3. デフォルトの設定

システム管理者は、GConf パスファイル内の `include` 命令を使って、別の設定ソースを指定できます。

含まれる設定ソース	説明
/etc/gconf/2/local-mandatory.path	この設定ソースは、特定のシステムの必須設定を格納するために使用します。
\$ (HOME) /.gconf.path	ユーザーは、設定ソースの場所をホームディレクトリ内の .gconf.path というファイルに指定します。
/etc/gconf/2/local-defaults.path	この設定ソースは、特定のシステムのデフォルトの設定値を格納するために使用します。

GConf パスファイル内の apoc 設定を使用して、Sun Java Desktop System Configuration Manager のバックエンドモジュールを指定します。

GConf スキーマ

「GConf スキーマ」は、「GConf スキーマキー」と「GConf スキーマオブジェクト」の総称です。以下の表は、スキーマキー、スキーマオブジェクト、およびこれらの項目と設定キーとの関連性について説明しています。

項目	説明
設定キー	アプリケーション設定に対応する GConf リポジトリ内の要素
スキーマキー	設定キーのスキーマオブジェクトを格納するキー。
スキーマオブジェクト	以下のような設定キー用の情報を含む設定ソース内の要素 ■ 設定キーを使用するアプリケーションの名前 ■ 設定キーに必要な値の型 (たとえば、int、bool など) ■ 設定キーのデフォルト値 ■ 設定キーについての簡単な記述

以下の表は、設定キー、スキーマキー、およびスキーマオブジェクトの例を示しています。

項目	例
設定キー	/desktop/gnome/interface/font_name
スキーマキー	/schemas/desktop/gnome/interface/font_name

項目	例
スキーマオブジェクト	<pre> <schema> <applyto>/desktop/gnome/interface/font_name</applyto> <key>/schemas/desktop/gnome/interface/font_name</key> <owner>gnome</owner> <type>string</type> <default>Sans 10</default> <locale name="C"> <short>Default font</short> <long>Name of the default font used by gtk+.</long> </locale> </schema> </pre>

設定キーにスキーマキーを関連付けることができます。たとえば、`/desktop/gnome/interface/font_name` キーは次のスキーマキーを含んでいます。

```

<entry name="font_name" mtime="1034873859"
schema="/schemas/desktop/gnome/interface/font_name"/>

```

設定キーにスキーマキーを関連付けると、設定は、そのスキーマキーのスキーマオブジェクト内で指定されている推奨値を使用します。推奨値は、スキーマオブジェクトの `<default>` 要素に含まれています。デフォルトでは、デフォルト設定ソース内のすべての設定キーが、スキーマキーと関連付けられます。

通常、スキーマはデフォルト設定ソースに格納されます。

GConf スキーマ定義ファイル

スキーマは、「スキーマ定義ファイル」から生成されます。スキーマ定義ファイルは、特定のアプリケーションにおけるすべてのキーの特性を定義します。スキーマ定義ファイルには、`.schemas` 拡張子が付きます。

スキーマ定義ファイルは、`/etc/gconf/schemas` ディレクトリに含まれています。スキーマ定義ファイルを使用して、新しい設定ソースを作成できます。

いくつかのスキーマ定義ファイルは、Java Desktop System ユーザーインタフェースの一部と緊密に対応しています。たとえば、`system_http_proxy.schemas` は、Internet 設定ツールに対応しています。ほかのスキーマ定義ファイルは、Java Desktop System ユーザーインタフェースにはない設定キーを含んでいます。たとえば、`/apps/panel/global/tooltips_enabled` キーはユーザーインタフェースにはありません。

Java Desktop System ユーザーインタフェースのいくつかは、複数のスキーマ定義ファイルの設定キーを表す設定を含んでいます。たとえば、Shortcuts 設定ツールは、`panel-global-config.schemas` ファイル および `metacity.schemas` ファイルのキーを表す設定を含んでいます。

GConf デーモン

GConf デーモンは、`gconfd-2` です。GConf デーモンは、設定の値が変更されると、アプリケーションに通知します。たとえば、「メニューとツールバー」設定ツールでツールバーの「アイコンのみ」を表示するように選択するとします。設定ツールでこのオプションを選択すると、直ちに、開いているすべてのアプリケーションのツールバーが更新されます。GConf デーモンは、ローカルでも、ネットワークを介してでも動作することができます。

GConf デーモンのインスタンスは、各ユーザーごとに起動されます。GConf デーモンは、認証やデータのセキュリティなどの複雑な問題を処理する必要がありません。GConf デーモンは、起動するときに GConf パスファイルを読み取ります。GConf デーモンは、アプリケーションと設定ソース間のすべてのアクセスを管理します。

アプリケーションが設定キーの値を要求すると、デーモンは次のようにして設定ソースを検索します。

1. パスファイルに指定された順に、各設定ソース内の設定キーの値を検索します。値が見つかると、その値を返します。
2. 値が見つからない場合は、パスファイルに指定された順に、各設定ソース内の設定キーに対応するスキーマキーを検索します。
3. スキーマキーが見つかると、スキーマキーの値を調べます。
4. スキーマキーの値がスキーマオブジェクトの場合、そのスキーマオブジェクトの `<default>` 要素内の推奨値を返します。

GConf デーモンは、設定キーの値をキャッシュに入れます。すべてのアプリケーションがこのキャッシュを使用するため、アプリケーションが設定ソースにアクセスするのは一度だけです。

GConf デーモンを終了するには、次のコマンドを実行します。

```
# gconftool-2 --shutdown
```

GConf コマンドラインツール

GConf は、コマンドラインツールの `gconftool-2` を含んでいます。`gconftool-2` コマンドを使用して、次の作業を実行できます。

- キーの値を設定する
- キーの値を表示する
- アプリケーションをインストールするときに、スキーマ定義ファイルからスキーマをインストールする

たとえば、次のコマンドを使用して /desktop/gnome ディレクトリおよびサブディレクトリ内のすべてのキー値を表示することができます。

gconftool-2 --recursive-list /desktop/gnome

表 1-1 は、gconftool-2 コマンドに使用できるオプションを示しています。

表 1-1 gconftool-2 コマンドオプション

オプション	機能
--all-dirs	指定したディレクトリ内のすべてのサブディレクトリをリストする
--all-entries	指定したディレクトリ内のすべてのキー値を表示する
--config-source= configuration-source	--direct オプションと共に使用して、使用する設定ソースを指定する。このオプションで設定ソースを指定しない場合、パスファイル内のすべての設定ソースでコマンドが実行される
--direct	--config-source オプションと共に使用して、設定ソースに直接アクセスする。このオプションを使用する場合、GConf はサーバーを省略する。このオプションを使用する前に、GConf デーモンの gconfd-2 が実行されていないことを確認する必要がある
--dump	ユーザーが指定した GConf リポジトリディレクトリ内に、すべての設定キーを含む一覧を生成する。一覧には、すべてのキーの XML 記述が含まれる。一覧は、 <gconfentryfile> 要素内に含まれている。 たとえば、このオプションからの出力をリダイレクトして、ユーザーのパネル設定に関係のあるキーをすべて含むファイルを生成できる。このファイルでは --load オプションが使用できる。
--get	指定した設定キーの値を表示する。また、指定したスキーマキーのスキーマオブジェクト内の要素の値も表示する
--help	gconftool-2 コマンドに関するヘルプメッセージと、gconftool-2 コマンドに使用できるオプションを表示する
--load=filename	設定ソースの現在のディレクトリ内にある設定キーの値を、指定したファイルの値に設定できる。 <gconfentryfile> 要素にキーの XML 記述を含むファイルを指定すること
--long-desc=description	--set-schema オプションと共に使用して、スキーマキーの長い説明を指定する
--makefile-install-rule	スキーマ定義ファイルをアプリケーションにインストールする

表 1-1 gconftool-2 コマンドオプション (続き)

オプション	機能
<code>--owner=owner</code>	<code>--set-schema</code> オプションと共に使用して、スキーマキーの所有者を指定する
<code>--recursive-list</code>	指定したディレクトリ内のすべてのサブディレクトリにあるすべての設定キーの値を表示する
<code>--recursive-unset</code>	ディレクトリ内のすべてのサブディレクトリで、ユーザー設定からデフォルト設定ソースの設定までを含む、すべての設定キーの値をリセットする
<code>--set</code>	設定キーの値を設定し、その値をユーザー設定ソースに書き込む。<code>--type</code> オプションは <code>--set</code> オプションと共に使用して、設定する値のデータ型を指定する。たとえば、以下のコマンドは、ユーザー設定ソースの <code>/apps/gnome-terminal/profiles/Default/background_color</code> キーの値を設定する <pre># gconftool-2 --set "/apps/gnome-terminal/profiles/Default /background_color" --type string "#000000"</pre> <code>--direct</code> オプションと <code>--config-source</code> オプションを <code>--set</code> オプションと共に使用して、値を別の設定ソースに書き込むこともできる
<code>--set-schema</code>	スキーマキーの属性の値を設定し、その値をデフォルト設定ソースに書き込む 次のオプションを <code>--set-schema</code> オプションと共に使用して、更新する属性を指定する ■ <code>--type</code> ■ <code>--short-desc</code> ■ <code>--long-desc</code> ■ <code>--owner</code> 次のコマンドは <pre>/apps/gnome-terminal/profiles/Default /background_color</pre> キーに対して、スキーマキーの短い説明を設定する <pre># gconftool-2 --set-schema "/schemas/apps/gnome-terminal/profiles /Default/background_color" --short-desc "Default background color of terminal"</pre>
<code>--short-desc=description</code>	<code>--set-schema</code> オプションと共に使用して、スキーマキーの短い説明を指定する
<code>--shutdown</code>	GConf デーモンを終了する

表 1-1 gconftool-2 コマンドオプション (続き)

オプション	機能
<code>--type=data-type</code>	設定キーの値を設定する場合にデータ型を指定する。スキーマキーの属性値を設定する場合にもこのオプションを使用できる。以下のような有効なデータ型がある ■ <code>bool</code> ■ <code>float</code> ■ <code>int</code> ■ <code>list</code> ■ <code>pair</code> ■ <code>string</code>
<code>--unset</code>	ユーザー設定からデフォルト設定ソースの設定までを含む、設定キーの値をリセットする
<code>--usage</code>	<code>gconftool-2</code> コマンドに関する簡単なヘルプメッセージと、 <code>gconftool-2</code> コマンドに使用できるオプションを表示する

設定値の設定

設定キーに対して、必須値またはデフォルト値を設定できます。ユーザーの必須設定値またはデフォルト設定値を変更する前に、GConf デーモンがどのユーザーに対しても実行されていないことを確認する必要があります。また、すべてのユーザーがログアウトしていることを確認してから、ユーザーの設定値を変更してください。

設定キーの必須値またはデフォルト値を設定するには、`gconftool-2` コマンドを次のように使用します。

```
# gconftool-2 --direct --config-source configuration-source --type
data-type --set preference-key value
```

たとえば、`wwwproxy.xyz.com` を必須の HTTP プロキシホストとして設定するには、次のコマンドを実行します。

```
# gconftool-2 --direct --config-source
xml:readwrite:/etc/gconf/gconf.xml.mandatory --type string --set
/system/http_proxy/host wwwproxy.xyz.com
```

ユーザーは、この設定値を上書きできません。

`gconftool-2` コマンドを使用して、デフォルト値を設定することもできます。たとえば、ワークスペースのデフォルト数を 5 に設定するには、次のコマンドを実行します。

```
# gconftool-2 --direct --config-source
xml:readwrite:/etc/gconf/gconf.xml.defaults --type int --set
/apps/metacity/general/num_workspaces 5
```

ユーザーは、この設定値を上書きできます。



注意 – ユーザーの必須設定値またはデフォルト設定値を変更する前に、すべてのユーザーがログアウトしていることを確認してください。

一般的な環境の設定

この節では、必須値またはデフォルト値を一般的な設定に割り当てる方法を説明します。

HTTP プロキシの設定

HTTP プロキシを設定するには、`/system/http_proxy/` の設定キーの値を変更します。たとえば、HTTP プロキシホストに必須値を設定するには、次のコマンドを実行します。

```
# gconftool-2 --direct --config-source
xml:readwrite:/etc/gconf/gconf.xml.mandatory --type string --set
/system/http_proxy/host proxy-name
```

HTTP プロキシホストにデフォルト値を設定するには、次のコマンドを実行します。

```
# gconftool-2 --direct --config-source
xml:readwrite:/etc/gconf/gconf.xml.defaults --type string --set
/system/http_proxy/host proxy-name
```

ほかの HTTP プロキシ関連の環境を設定することもできます。ほかの HTTP プロキシ設定に関する情報については、`system_http_proxy.schemas` スキーマ定義ファイルを参照してください。

ワークスペース数の設定

ワークスペースの必須の数を設定するには、次のコマンドを使用します。

```
# gconftool-2 --direct --config-source
xml:readwrite:/etc/gconf/gconf.xml.mandatory --type int --set
/apps/metacity/general/num_workspaces integer
```

ワークスペースのデフォルトの数を設定するには、次のコマンドを使用します。

```
# gconftool-2 --direct --config-source
xml:readwrite:/etc/gconf/gconf.xml.defaults --type int --set
/apps/metacity/general/num_workspaces integer
```

ほかのウィンドウマネージャの環境を設定することもできます。ほかのウィンドウマネージャの設定に関する情報については、metacity.schemas スキーマ定義ファイルを参照してください。

キーボードショートカットの設定

キーボードショートカット設定を行うには、
/apps/metacity/global_keybindings/panel_run_dialog にある設定キーの値を変更します。たとえば、「アプリケーションの実行」ダイアログを開くときに、ユーザーが Alt + F3 キーボードショートカットのみを使用するように設定したい場合があります。この必須値を設定するには、次のコマンドを実行します。

```
# gconftool-2 --direct --config-source
xml:readwrite:/etc/gconf/gconf.xml.mandatory --type string --set
/apps/metacity/global_keybindings/panel_run_dialog '<Alt>F3'
```

ほかのキーボードショートカット環境を設定することもできます。ほかのキーボードショートカットの設定に関する情報については、metacity.schemas スキーマ定義ファイルを参照してください。

パネルとパネルオブジェクトの設定

panel-default-setup.entries ファイルは、Java Desktop System 内のパネルについて、次の内容を指定します。

- パネルの数
- パネルのタイプ
- パネルのプロパティ
- パネルの内容

パネルおよびパネルオブジェクトを個別に設定するのは、複雑な作業になります。パネルやパネルオブジェクトを個別に設定するには、まず panel-default-setup.entries ファイルの構造を理解する必要があります。panel-default-setup.entries ファイルについては、[24 ページの「パネルおよびパネルオブジェクトの個別指定」](#)を参照してください。

パネルおよびパネルオブジェクトの環境を個別に設定するには、設定ソース内の多数の設定値を設定する必要があります。パネル設定の値を設定する一番簡単な方法は、--dump および --load オプションを指定して、gconftool-2 コマンドを使用することです。パネルおよびパネルオブジェクトの設定方法の詳細については、[26 ページの「パネルおよびパネルオブジェクトの環境を個別に設定」](#)を参照してください。

パネルおよびパネルオブジェクトの個別指定

panel-default-setup.entries ファイルには、パネルおよびパネルの内容を指定する部分があります。また panel-default-setup.entries ファイルは、スキーマキーの値を指定します。panel-default-setup.entries ファイルは、`/etc/gconf/schemas` ディレクトリ内にあります。

panel-default-setup.entries ファイルの構造は、以下のとおりです。

1. ファイル内のすべてのキーのベースパスを指定する、entrylist 要素。たとえば、以下の要素 (panel-default-setup.entries からの引用) は、キーのベースパスとして `/apps/panel/default_setup` を指定します。

```
<entrylist base="/apps/panel/default_setup">
```

たとえば、キー `general/toplevel_id_list` が panel-default-setup.entries 内で参照されている場合、キーのフルパスは `/apps/panel/default_setup/general/toplevel_id_list` です。

2. デスクトップのパネル、アプレット、およびその他のパネルオブジェクトの一般的な構造を指定するキー。以下のキーが、デスクトップ内に表示されるパネル、パネルオブジェクト、およびアプレットの数を指定します。

- `general/toplevel_id_list`
- `general/object_id_list`
- `general/applet_id_list`

これらのキーは、各パネル、パネルオブジェクト、およびアプレットに識別子も割り当てます。たとえば、以下の panel-default-setup.entries のサンプルでは、デスクトップに1つのパネルが表示されるように指定しています。

```
<entry>
  <key>general/toplevel_id_list</key>
  <schema_key>/schemas/apps/panel/general/toplevel_id_list</schema_key>
  <value>
    <list type="string">
      <value>
        <string>bottom_panel</string>
      </value>
    </list>
  </value>
</entry>
```

panel-default-setup.entries ファイルでは、識別子 `bottom_panel` が画面の最下部のパネルを識別します。

3. パネルのプロパティを指定するキー。パネル設定キーは、次のように設定されています。

`toplevels/panel-name/panel-property-key`

たとえば、`toplevels/bottom_panel/size` のキーはボトムパネルのサイズを指定します。

4. パネルオブジェクト、パネルオブジェクトプロパティ、およびオブジェクトが存在するパネルを指定するキー。たとえば、以下の `panel-default-setup.entries` のサンプルでは、メインメニューオブジェクトが ボトムパネルの左側に表示されます。

```
<entry>
  <key>objects/main_menu/object_type</key>
  <schema_key>/schemas/apps/panel/objects/object_type</schema_key>
  <value>
    <string>menu-object</string>
  </value>
</entry>
<entry>
  <key>objects/main_menu/toplevel_id</key>
  <schema_key>/schemas/apps/panel/objects/toplevel_id</schema_key>
  <value>
    <string>bottom_panel</string>
  </value>
</entry>
<entry>
  <key>objects/main_menu/position</key>
  <schema_key>/schemas/apps/panel/objects/position</schema_key>
  <value>
    <int>0</int>
  </value>
</entry>
```

5. アプレット、アプレット設定、およびアプレットが存在するパネルを指定するキー。たとえば、以下の `panel-default-setup.entries` のサンプルでは、Window List アプレットがボトムパネルに表示されます。

```
<entry>
  <key>applets/window_list/object_type</key>
  <schema_key>/schemas/apps/panel/objects/object_type</schema_key>
  <value>
    <string>bonobo-applet</string>
  </value>
</entry>
<entry>
  <key>applets/window_list/toplevel_id</key>
  <schema_key>/schemas/apps/panel/objects/toplevel_id</schema_key>
  <value>
    <string>bottom_panel</string>
  </value>
</entry>
<entry>
  <key>applets/window_list/position</key>
  <schema_key>/schemas/apps/panel/objects/position</schema_key>
  <value>
```

```

        <int>3</int>
    </value>
</entry>
.
.
.
<entry>
  <key>applets/window_list/bonobo_iid</key>
  <schema_key>/schemas/apps/panel/objects/bonobo_iid_type</schema_key>
  <value>
    <string>OAFIID:GNOME_WindowListApplet</string>
  </value>
</entry>

```

OAFIID は、アプレットの一意の識別子です。特定のアプレットの OAFIID を検索するには、`/usr/lib/bonobo/servers` ディレクトリ内にあるそのアプレットの `.server` ファイルを参照します。たとえば、以下の `GNOME_Wncklet_Factory.server` の抜粋では、Window List アプレットの OAFIID が示されます。

```

<oaf_server iid="OAFIID:GNOME_WindowListApplet"
type="factory" location="OAFIID:GNOME_Wncklet_Factory">

```

パネルおよびパネルオブジェクトの環境を個別に設定

パネルやパネル内のオブジェクトの設定を行うには、次の手順を実行します。

1. Java Desktop System セッションにログインし、必要に応じてパネルを設定します。
2. `--dump` オプションを指定して `gconftool-2` コマンドラインツールを使用し、使用しているパネル設定の XML 記述を含むファイルを生成します。`--dump` オプションは、ユーザーが指定した GConf リポジトリディレクトリ内にあるすべての設定キーを含む一覧を生成します。

たとえば、以下のコマンドはデフォルトパネル設定の XML 記述を `my-panel-setup.entries` というファイルに作成します。

```

# gconftool-2 --dump /apps/panel/profiles/default >
my-panel-setup.entries

```

3. `my-panel-setup.entries` ファイルをテキストエディタで開き、必要に応じて修正します。

たとえば、デスクトップエントリファイルの場所を変更します。以下に、`--dump` オプションで生成されたファイルの抜粋を示します。

```
<entry>
  <key>objects/object_16/launcher_location</key>
  <schema_key>/schemas/apps/panel/objects/launcher_location</schema_key>
  <value>
    <string>hadjaha-00adce02f7.desktop</string>
  </value>
</entry>
```

上のサンプルで、hadjaha-00adce02f7.desktop への参照を、広く使用可能な別のデスクトップエントリファイルに変更することも考えられます。

--dump オプションでパネル設定を生成すると、パネルオブジェクトの位置は絶対的な位置になります。パネルオブジェクトの位置を絶対的な位置から相対的な位置に変更したい場合もあります。パネルの一番左端のオブジェクトの position 値は 0 です。その次のオブジェクトの position 値は 1 という具合になっています。オブジェクトをパネルの右側と相対する位置に置くには、right_stick キーを true に設定します。

4. --load オプションを指定して gconftool-2 コマンドラインツールを使用し、デフォルトの設定ソースの値を my-panel-setup.entries ファイル内の値に設定します。たとえば、以下のコマンドは、デフォルトの設定ソースのキー値を my-panel-setup.entries 内の対応するキーの値に設定します。

```
# gconftool-2 --direct --config-source
xml:readwrite:/etc/gconf/gconf.xml.defaults --load
my-panel-setup.entries
```

ルック&フィールの設定

この節では、ルック&フィールの設定に必須値またはデフォルト値を割り当てる方法を説明します。

フォントの設定

フォントを設定するには、2つの設定キーの値を変更します。以下の表は、変更するキーと、そのキーに対応するユーザーインタフェース部分を示しています。

GConf の場所	ユーザーインタフェースコンポーネント
/desktop/gnome/interface/font_name	フォント設定ツールの「アプリケーション用フォント」オプション

GConf の場所	ユーザーインタフェースコンポーネント
/apps/nautilus/preferences/desktop_font	フォント設定ツールの「デスクトップ用フォント」オプション

たとえば、Sans 12 を必須アプリケーション用フォントとして設定するには、次のコマンドを実行します。

```
# gconftool-2 --direct --config-source
xml:readwrite:/etc/gconf/gconf.xml.mandatory --type string --set
/desktop/gnome/interface/font_name "Sans 12"
```

palatino 12 をデフォルトのデスクトップオブジェクトフォントとして設定するには、次のコマンドを実行します。

```
# gconftool-2 --direct --config-source
xml:readwrite:/etc/gconf/gconf.xml.defaults --type string --set
/apps/nautilus/preferences/desktop_font "palatino 12"
```

背景の設定

デスクトップ背景の設定を行うには、/desktop/gnome/background にある設定キーの値を変更します。たとえば、背景に必須イメージを設定するには、次のコマンドを実行します。

```
# gconftool-2 --direct --config-source
xml:readwrite:/etc/gconf/gconf.xml.mandatory --type string --set
/desktop/gnome/background/picture_filename filename.png
```

この設定にデフォルト値を設定するには、次のコマンドを実行します。

```
# gconftool-2 --direct --config-source
xml:readwrite:/etc/gconf/gconf.xml.defaults --type string --set
/desktop/gnome/background/picture_filename filename.png
```

ほかの背景を設定することもできます。ほかの背景設定に関する情報については、desktop_gnome_background.schemas スキーマ定義ファイルを参照してください。

スプラッシュイメージの設定

スプラッシュイメージを設定するには、/apps/gnome-session/options/ の設定キーの値を変更します。たとえば、スプラッシュイメージを表示しないように設定するには、必須値を次のように設定します。

```
# gconftool-2 --direct --config-source
xml:readwrite:/etc/gconf/gconf.xml.mandatory --type bool --set
/apps/gnome-session/options/show_splash_screen false
```

この設定にデフォルト値を設定するには、次のコマンドを実行します。

```
# gconftool-2 --direct --config-source  
xml:readwrite:/etc/gconf/gconf.xml.defaults --type bool --set  
/apps/gnome-session/options/show_splash_screen false
```

ほかのスプラッシュイメージを設定することもできます。ほかのスプラッシュイメージ設定に関する情報については、`gnome-session.schemas` スキーマ定義ファイルを参照してください。

デフォルト設定値の復元

ユーザーのデフォルト設定値を復元するには、次のコマンドを実行します。

```
# gconftool-2 --direct --config-source user-configuration-source  
--recursive-unset
```

`user-configuration-source` は、ユーザーのホームディレクトリにある `.gconf` ディレクトリ内の設定ソースに置き換えます。

このコマンドは、すべてのサブディレクトリで、ユーザー設定からデフォルト設定ソースの設定までを含むすべての設定キーの値をリセットします。

第 2 章

メニューのカスタマイズ

この章では、Java Desktop System でどのようにメニューが実装されるか、また、管理者はどのようにメニューをカスタマイズできるかを説明します。

メニューの概要

Java Desktop System でどのようにメニューが実装されるかにより、次のことが可能になります。

- 簡単にメニュー階層をカスタマイズできる。メニュー階層は、ファイルシステムの階層構造に基づいていません。メニュー階層は、少数のファイルを編集するだけでカスタマイズできます。アプリケーションを変更したり、ファイルを移動する必要はありません。
- 簡単にアプリケーションをインストールできる。アプリケーションをインストールするときに、メニュー階層に関する情報をアプリケーションに提供する必要はありません。
- ユーザーがメニューを変更できないように設定する。

Java Desktop System 上のメニューでは、次のコンポーネントを使用します。

- ファイル抽象レイヤー
- vfolder
- デスクトップエントリファイル
- ディレクトリエントリファイル

ファイル抽象レイヤー

gnome-vfs ファイル抽象レイヤーは、アプリケーションがファイルと相互に作用するための単純かつ一般化された方法を提供します。また、ファイル抽象レイヤーは、特定のメニュー構成ファイルに対応付ける *URI (Uniform Resource Identifier)* ロケーションを提供します。

すべてのユーザー用にメニューまたはメニュー項目を追加するには、そのメニューまたはメニュー項目を URI ロケーションの 1 つに追加する必要があります。項目を追加できるメニューと、そのメニューに対応する URI ロケーションは表 2-1 に示しています。

表 2-1 システム管理者用のメニューと URI ロケーション

メニュー	URI ロケーション
すべてのユーザーの「アプリケーション」メニュー	applications-all-users:///
すべてのユーザーの「設定」メニュー	preferences-all-users:///

メニューまたはメニュー項目をユーザー自身のシステムにだけ追加する場合は、それらのメニューまたはメニュー項目を他の URI ロケーションに追加する必要があります。表 2-2 は、ユーザーが項目を追加できるメニューと、それらのメニューに対応する URI ロケーションを示しています。

表 2-2 ユーザー用のメニューと URI ロケーション

メニュー	URI ロケーション
「アプリケーション」メニュー	applications:///
「設定」メニュー	preferences:///

ユーザーによるメニューのカスタマイズ方法については、『*Java Desktop System Release 2 ユーザーズガイド*』の「メニューの操作方法」を参照してください。

vfolder とメニュー

一般的に、「vfolder」は、システム上の物理的な場所 (1 箇所または複数) に存在する項目の仮想表現になります。たとえば、vfolder が、いくつかのディレクトリの内容を表す場合があります。また vfolder は、1 つまたは複数の物理的な場所の抽象概念になります。Java Desktop System のメニューの場合、vfolder は、物理的に複数のディレクトリに置かれている項目のメニュー上の表現になります。

vfolder 情報ファイルは、vfolder を説明する XML ファイルです。vfolder 情報ファイルは、メニューの構造を指定します。また、メニューの名前と、メニューに表示されるアプリケーションの順序も指定します。vfolder 情報ファイルには、.vfolder-info ファイル拡張子が付きます。

以下に vfolder 情報ファイルのサンプルを示します。

```
<VFolderInfo>
.
.
.
  <Folder>
    <Name>Applications</Name>
    <Desktop>Applications.directory</Desktop>
    <Folder>
      <Name>Accessories</Name>
      <Desktop>Accessories.directory</Desktop>
      <Query>
        <And>
          <Keyword>Application</Keyword>
          <Keyword>Utility</Keyword>
        </And>
      </Query>
      <DontShowIfEmpty/>
    </Folder>
  .
  .
  .
</Folder>
</VFolderInfo>
```

表 2-3 は、vfolder 情報ファイル内の要素を示しています。

表 2-3 vfolder 情報ファイルの要素

要素	説明
<Folder>	メニューの名前、内容、構造を定義する要素が含まれる
<Name>	メニューの名前を指定する。
<Desktop>	メニューの名前、コメント、アイコンを指定するディレクトリエントリファイルの名前

表 2-3 vfolder 情報ファイルの要素 (続き)

要素	説明
<Query>	デスクトップエントリファイルに対するクエリー。クエリー要件と一致するデスクトップエントリファイルと、メニュー項目がメニュー上に表示される 上記抜粋内のクエリーでは、Categories キー内にキーワード Application と Utility を含むデスクトップエントリファイルが検索される。一致するデスクトップエントリファイルが「アプリケーション」メニューに表示される この要素は任意
<DontShowIfEmpty/>	この要素がある場合は、項目を何も含まないメニューは表示されない この要素は任意

デスクトップエントリファイル

「デスクトップエントリファイル」は、メニュー内の項目に関する情報を提供するデータファイルです。デスクトップエントリファイルは、名前、実行するコマンド、アイコンなどの項目の詳細を指定します。また、デスクトップエントリファイルには、メニュー階層内での項目の位置を決定するキーワードが含まれています。デスクトップエントリファイルには、.desktop ファイル拡張子が付きます。

以下は、デスクトップエントリファイルのサンプルです。

```
[Desktop Entry]
Encoding=UTF-8
Name=Calculator
Comment=Perform calculations
Exec=gcalc-tool
Icon=gcalc-tool.png
Terminal=false
Type=Application
Categories=GNOME;Application;Utility;
```

表 2-4 は、デスクトップエントリファイルで最も重要なキーを説明しています。

表 2-4 デスクトップエントリキー

デスクトップエントリキー	説明
Encoding	デスクトップエントリファイルのエンコーディングを指定する

表 2-4 デスクトップエントリキー (続き)

デスクトップエントリキー	説明
Name	項目の名前を指定する。この名前は、メニュー上の項目として表示される
Comment	項目の短い説明を指定する。コメントは、メニュー上のメニュー項目を指したときにツールチップとして表示される
Exec	メニューから項目を選択したときに実行されるコマンドを指定する
Icon	項目を表すアイコンのファイル名を指定する。ファイル名へのパス、あるいはファイル拡張子は指定しない
Terminal	Exec キーで指定したコマンドを端末で実行するかどうかを指定する。値が true の場合は、コマンドが端末で実行される コマンドを実行するウィンドウを作成しない場合は、このキーの値を true にする必要がある
Type	項目のタイプを指定する。次のいずれかを指定する ■ Application: アプリケーションを起動する項目の場合はこのオプションを入力 ■ Link: ファイル、フォルダ、または URI にリンクする項目の場合はこのオプションを入力
Categories	項目を説明するキーワードを指定する。キーワードはセミコロン (;) で区切る。標準のカテゴリキーワードのリストについては、以下の URL のデスクトップメニュー仕様を参照 http://www.freedesktop.org vfolder 情報ファイルは、メニューに入力するキーワードを対応付ける

デスクトップエントリファイルのキーに関する詳細については、以下の URL のデスクトップエントリ仕様を参照してください。

<http://www.freedesktop.org>

注 - パネルランチャーやデスクトップオブジェクトも、デスクトップエントリファイルを使用します。ランチャーやデスクトップオブジェクト用のデスクトップエントリファイルは、メニュー項目用のデスクトップエントリファイルと同じ情報を提供します。たとえば、デスクトップエントリファイルは、ユーザーがランチャーまたはオブジェクトを選択したときに実行されるコマンドを提供します。

ディレクトリエントリファイル

「ディレクトリエントリファイル」は、メニューに関する情報を提供するデータファイルです。「ディレクトリエントリファイル」は、名前、ツールチップ、アイコンなどのメニューの詳細を指定します。ディレクトリエントリファイルには、`.directory` ファイル拡張子が付きます。

以下は、ディレクトリエントリファイルのサンプルです。

```
[Desktop Entry]
Name=Accessories
Comment=Accessories menu
Icon=gnome-util.png
Type=Directory
```

表 2-5 は、ディレクトリエントリファイルの最も重要なキーを説明しています。

表 2-5 ディレクトリエントリキー

ディレクトリエントリキー	説明
Name	メニューの名前を指定する。この名前は、メニューに表示される
Comment	メニューの短い説明を指定する。コメントは、メニューを指したときにツールチップとして表示される
Icon	メニューを表すアイコンのファイル名を指定する。ファイル名へのパス、あるいはファイル拡張子は指定しない
Type	メニューのタイプを指定する。このキーの値は常に <code>Directory</code>

メニューの編集

以下の Java Desktop System コンポーネントを使用してメニューを編集できます。

- Nautilus ファイルマネージャ
- パネル上のメニュー

ファイルマネージャを使用してすべてのユーザーにメニューまたはメニュー項目を追加する場合は、メニューまたはメニュー項目を URI ロケーションに追加します。項目を追加できるメニューと、そのメニューに対応する URI ロケーションは表 2-1 に示しています。

パネルを使用して、すべてのユーザーのメニューをカスタマイズする場合は、メニュー項目ポップアップメニューを使用します。詳細は、『*Java Desktop System Release 2 ユーザーズガイド*』の「メニューの操作方法」を参照してください。

メニューのカスタマイズに、メニュー構成ファイルおよびメニューデータファイルを使用することもできます。

メニューの追加

次の方法を使用して、メニューをすべてのユーザーに追加できます。

- ファイルマネージャを使用する
- メニュー構成ファイルおよびメニューデータファイルを変更する

ファイルマネージャによるメニューの追加

すべてのユーザーにメニューを追加するには、次の手順を実行します。

1. ファイルマネージャウィンドウで、メニューを追加する場所にアクセスします。たとえば、「アプリケーション」メニューにメニューを追加する場合は、「場所」フィールドに `applications-all-users:///` と入力して、リターンキーを押します。
2. 「ファイル」→「新規フォルダ」を選択します。タイトルのないフォルダ「未タイトルのフォルダ」が表示区画に追加されます。フォルダの名前が選択されています。

注 - Solaris プラットフォームでは、フォルダの名前を入力する前にファイルマネージャウィンドウの表示を更新する必要があります。

3. フォルダの名前を入力して、Return キーを押します。手順1 でアクセスした場所用の `vfolder` 情報ファイルは、新しいメニューの詳細を反映して自動的に更新されます。フォルダの名前は、メニューの名前として表示されます。

次回ユーザーがログインすると、指定した場所にメニューが表示されます。

メニューファイルによるメニューの追加

すべてのユーザーにメニューを追加するには、次の手順を実行します。

1. 追加する項目用にディレクトリエントリファイルを作成します。ディレクトリエントリファイルは、`/usr/share/gnome/vfolders` ディレクトリに作成します。ディレクトリエントリファイルの詳細については、[36 ページの「ディレクトリエントリファイル」](#)を参照してください。
2. メニューを追加する場所用の `vfolder` 情報ファイルの場所を確認します。「アプリケーション」メニューにメニューを追加する場合は、ファイル `/etc/gnome-vfs-2.0/vfolders/applications-all-users.vfolder-info` の場所を確認します。

3. vfolder 情報ファイルに、新しいメニュー用の <Folder> 要素を追加します。
vfolder 情報ファイルの詳細については、32 ページの「vfolder とメニュー」を参照してください。

次回ユーザーがログインすると、指定した場所にメニューが表示されます。

メニューへの項目の追加

すべてのユーザーのメニューに項目を追加するには、次の手順を実行します。

1. 追加する項目用にデスクトップエントリファイルを作成します。デスクトップエントリファイルの詳細については、34 ページの「デスクトップエントリファイル」を参照してください。
2. ファイルマネージャウィンドウを開きます。「ファイル」→「新規ウィンドウ」を選択して、別のファイルマネージャウィンドウを開きます。
3. 1つのウィンドウで、メニュー項目を追加する場所にアクセスします。たとえば、「設定」メニューにメニュー項目を追加する場合は、「場所」フィールドに `preferences-all-users:///` と入力し、リターンキーを押します。
4. 別のウィンドウで、そのメニュー項目用に作成したデスクトップエントリファイルを選択します。メニュー項目を追加する場所にデスクトップエントリファイルをドラッグします。

あるいは、デスクトップエントリファイルをコピーして、メニュー項目を追加する場所にそのファイルを貼り付けます。

次回ユーザーがログインすると、指定した場所にメニュー項目が表示されます。

メニューの設定を編集する

すべてのユーザーのメニューの設定を編集するには、次の手順を実行します。

1. パネルから編集するメニューを開きます。メニュー上の任意の項目を右クリックします。
2. 「メニュー全体」→「設定」を選択します。「ランチャーの設定」ダイアログが表示されます。
3. 「ランチャーの設定」ダイアログでメニューの設定を変更します。「ランチャーの設定」ダイアログの要素については、『*Java Desktop System Release 2 ユーザーズガイド*』の「パネルの操作方法」を参照してください。
4. 「了解」をクリックします。

メニュー項目を編集する

メニュー項目を編集するには、次の手順を実行します。

1. パネルから、編集する項目を含むメニューを開きます。編集する項目を右クリックします。
2. 「設定」を選択します。「ランチャーの設定」ダイアログが表示されます。
3. 「ランチャーの設定」ダイアログでメニュー項目の設定を変更します。「ランチャーの設定」ダイアログの要素については、『*Java Desktop System Release 2 ユーザーズガイド*』の「パネルの操作方法」を参照してください。
4. 「了解」をクリックします。

メニューから項目を削除する

メニューから項目を削除するには、パネルから削除する項目を含むメニューを開きます。削除する項目を右クリックします。「ランチャーをメニューから削除」を選択します。

次回ユーザーがログインすると、そのメニュー項目はメニューに表示されません。

ユーザーが変更できないメニューの設定

以下の条件が当てはまる場合、ユーザーはメニューを変更できません。

- メニューに対応する vfolder 情報ファイルが `/etc/gnome-vfs-2.0/vfolders` ディレクトリに存在する
- vfolder 情報ファイルの名前が、メニューに対応する URI ロケーションの名前と同じである

ユーザーが変更できないようにメニューを設定するには、次の手順を実行します。

1. `/etc/gnome-vfs-2.0/vfolders` ディレクトリに、設定するメニュー用の vfolder 情報ファイルを作成します。
2. vfolder 情報ファイルの名前に、設定するメニューに対応する URI ロケーションの名前を指定します。たとえば、「アプリケーション」メニューを設定する場合、`/etc/gnome-vfs-2.0/vfolders` ディレクトリに `applications.vfolder-info` という名前で vfolder 情報ファイルを作成します。

第 3 章

テーマのインストール

この章では、テーマ、GNOME デスクトップで利用可能なテーマ設定、テーマ設定のオプションのインストール方法、およびカスタムオプションの作成方法について説明します。

テーマの概要

「テーマ」は、GNOME デスクトップの外観を指定する統合的な設定グループです。テーマを選ぶことにより、GNOME デスクトップの外観を変更できます。

テーマには、以下のように GNOME デスクトップの個別の部分を制御できる設定が含まれています。

コントロール	テーマのコントロール設定は、ウィンドウ、パネル、アプレットの外観を決定します。また、ウィンドウ、パネル、およびアプレット上に現れる GNOME 対応インタフェース (メニュー、アイコン、ボタンなど) の外観も決定します。利用可能なコントロール設定オプションの中には、特別なアクセシビリティ要件を満たすように設計されているものもあります。コントロール設定のオプションは、テーマ設定ツールの「コントロール」タブセクションで選択できます。
ウィンドウフレーム	テーマのウィンドウフレーム設定は、ウィンドウのフレームの外観だけを指定します。ウィンドウボーダー設定のオプションは、テーマ設定ツールの「ウィンドウボーダー」タブセクションで選択できます。
アイコン	テーマのアイコン設定は、パネルとデスクトップ背景にあるアイコンの外観を指定します。アイコン設定のオプションは、テーマ設定ツールの「アイコン」タブセクションで選択できます。

テーマのインデックスファイル

各テーマには、そのテーマの特徴を決めるインデックスファイルがあります。インデックスファイルの名前は、`/usr/share/themes/theme-name/index.theme` となります。

以下に、テーマのインデックスファイルのサンプルを示します。

```
[Desktop Entry]
Type=X-GNOME-Metatheme
Name=High Contrast Large
Name[es]=Alto contraste grande
Comment=Large black-on-white text and icons
Comment[es]=Textos e iconos grandes en negro sobre blanco
Encoding=UTF-8

[X-GNOME-Metatheme]
GtkTheme=HighContrastLargePrint
IconTheme=HighContrast
MetacityTheme=Atlanta
ApplicationFont=sans 18
```

以下の表に、テーマのインデックスファイルのキーを示します。

表 3-1 テーマのインデックスファイルのキー

インデックスファイルのキー	説明
Type	テーマが、コントロール、ウィンドウボーダー、アイコンなどの複数のテーマオプションの外観を決定することを指定する
Name	テーマの名前。テーマ設定ツールで表示されるテーマの名前
Comment	テーマの簡単な説明。テーマ設定ツールで、テーマの名前の下に表示されるテキスト
GtkTheme	テーマ設定ツールのコントロール設定に対応。ウィンドウ、パネル、アプレットに適用されるコントロール設定オプションを指定する
IconTheme	テーマ設定ツールのアイコン設定に対応。パネルとデスクトップ背景に適用されるアイコン設定オプションを指定する

表 3-1 テーマのインデックスファイルのキー (続き)

インデックスファイルのキー	説明
MetacityTheme	テーマ設定ツールのウィンドウフレーム設定に対応。ウィンドウに適用されるウィンドウフレーム設定のオプションを指定する
ApplicationFont	フォント設定ツールのアプリケーションフォント設定に対応

新しいコントロールオプションをインストールする

コントロール設定の新しいオプションをテーマ設定ツールで追加できます。コントロールオプションは、`/usr/share/themes` ディレクトリにあります。ファイルシステムでのコントロールオプションの典型的なディレクトリ構造は、以下のとおりです。

オプションファイル `/usr/share/themes/option-name/gtk-2.0/gtkrc`

イメージファイル `/usr/share/themes/option-name/pixmaps/*.*`

一般的にコントロール設定の新しいオプションは、`.tar.gz` ファイルの形で提供されます。新しいコントロールオプションをインストールするには、`.tar.gz` ファイルを解凍して、`.tar` ファイルを `/usr/share/themes` ディレクトリに展開します。

ユーザーは、コントロール設定に対してユーザー定義のオプションもインストールできます。コントロール設定のオプションをインストールすると、そのオプションは `$HOME/.themes` ディレクトリに保存されます。

新しいウィンドウフレームのオプションをインストールする

ウィンドウボーダー設定の新しいオプションをテーマ設定ツールで追加できます。ウィンドウフレームのオプションは、

`/usr/share/themes/option-name/metacity-1` ディレクトリにあります。ファイルシステムでのウィンドウフレームオプションの典型的なディレクトリ構造は、以下のとおりです。

オプションファイル /usr/share/themes/*option-name*
 /metacity-1/metacity-theme-1.xml

イメージファイル /usr/share/themes/*option-name*/metacity-1/*.*

一般的にウィンドウボーダー設定の新しいオプションは、.tar.gz ファイルの形で提供されます。新しいウィンドウフレームのオプションをインストールするには、.tar.gz ファイルを解凍して、.tar ファイルを /usr/share/themes ディレクトリに展開します。

ユーザーは、ウィンドウフレーム設定に対してユーザー定義のオプションもインストールできます。ウィンドウフレーム設定のオプションをインストールすると、そのオプションは \$HOME/.themes ディレクトリに保存されます。

新しいアイコンオプションをインストールする

アイコン設定の新しいオプションをテーマ設定ツールで追加できます。アイコンオプションは、/usr/share/icons/*option-name* ディレクトリにあります。ファイルシステムでのアイコンオプションの典型的なディレクトリ構造は、以下のとおりです。

オプションファイル /usr/share/icons/*option-name*

イメージファイル /usr/share/icons/*option-name*/icons/*.*

一般的にアイコン設定の新しいオプションは、.tar.gz ファイルの形で提供されます。新しいアイコンオプションをインストールするには、.tar.gz ファイルを解凍して、.tar ファイルを /usr/share/icons ディレクトリに展開します。

ユーザーは、アイコン設定に対してユーザー定義のオプションもインストールできます。アイコン設定のオプションをインストールすると、そのオプションは \$HOME/.icons/*option-name* ディレクトリに保存されます。

テーマ用の新しいアイコンのインストール

GNOME デスクトップには、特別な視覚的要件を持つユーザー用に設計されたいくつかのテーマがあります。たとえば、視力の弱いユーザー用に設計されたテーマが用意されています。アイコンを各テーマで正しく表示するためには、アイコンのバージョンが複数必要な場合もあります。

アプリケーション用に新しいアイコンをインストールしなければならない場合があります。新しいアイコンをインストールするときは、そのアイコンに対していくつかのバージョンを作成する必要があります。それによって、アイコンはそれぞれのテーマで正しく表示されます。以下のタイプのアイコンは、複数のバージョンを作成する必要があります。

- 「メインメニュー」内、またはファイルマネージャ内のアプリケーションを表すために使用されるアイコン
- GTK+ アプリケーションによって内部で使用されるアイコン、または GTK+ ストックアイコン

新しいアイコンをインストールするときは、そのアイコンに対していくつかのバージョンを作成する必要があります。それによって、アイコンはそれぞれのテーマで正しく表示されます。アイコンは、いくつかの形式で作成できます。たとえば PNG (Portable Network Graphic) 形式など。デスクトップ環境用のアイコンの推奨サイズは、48 x 48 ピクセルです。このサイズであれば、ほとんどのテーマでそのアイコンの大きさを変更できます。

新しいアイコンをインストールするときは、以下に示すアイコンを 48 x 48 ピクセルバージョンで作成します。

- 通常のアイコン
- 低コントラストのアイコン
- 高コントラストのアイコン
- 反転の高コントラストのアイコン

アイコンは、43 ページの「[新しいコントロールオプションをインストールする](#)」で説明されているイメージファイルの場所にインストールします。たとえば、アイコンを HighContrastLargePrint テーマに追加する場合は、そのアイコンを `/usr/share/themes/HighContrastLargePrint/pixmaps` ディレクトリに追加します。該当するテーマファイルにアイコンの参照先を追加します。たとえば、アイコンを HighContrastLargePrint テーマに追加する場合には、そのアイコンへの参照を `/usr/share/themes/HighContrastLargePrint/gtk-2.0/gtkrc` ファイルに追加します。テーマがアイコンを GTK ストックアイコン識別子に関連付けられるように `gtkrc` ファイルを修正します。

アプリケーションランチャー用およびパネル用のアイコンの作成方法については、<http://www.freedesktop.org/Standards/icon-theme-spec> を参照してください。

カスタムコントロールオプションを作成する

コントロール設定のオプションが、ユーザーのニーズに合わない場合は、カスタムコントロールオプションを作成できます。カスタムコントロールオプションを作成するには、次の手順を実行します。

1. オプションのディレクトリ構造を `/usr/share/themes` ディレクトリ内に作成します。ほかのオプションと同じディレクトリ構造を使用します。たとえば、`SmallPrint` というオプションを作成する場合は、次のディレクトリを作成します。
 - `/usr/share/themes/SmallPrint`
 - `/usr/share/themes/SmallPrint/gtk-2.0`
2. ユーザー要件をほぼ満たす `gtkrc` ファイルを見つけます。そのファイルを新しいオプションの `gtk-2.0` ディレクトリにコピーします。
3. テキストエディタを使用して、`gtkrc` ファイルを開き、必要に応じてインタフェース要素の属性を変更します。
4. 新しいオプションにイメージが含まれる場合は、そのイメージを新しいオプションの `pixmap` ディレクトリにインストールします。新しいオプションが別のオプションのイメージを使用する場合は、新しいオプション用にイメージのコピーを作成する必要はありません。ただし、`gtkrc` ファイルの `pixmap_path` エントリに登録されているイメージへの参照が正しいことを確認する必要があります。

ユーザーは、コントロール設定向けの新しいオプションを選択できるようになりました。

`gtkrc` ファイルについては、<http://developer.gnome.org/doc/API/2.0/gtk/gtk-Resource-Files.html> を参照してください。

第 4 章

フォントの設定

この章では、GNOME デスクトップでのフォントの設定方法について説明します。

フォント設定の概要

GNOME デスクトップは `fontconfig` フォント設定およびカスタマイズライブラリを使用します。`fontconfig` ライブラリでは、PostScript Type 1 フォントおよび TrueType フォントを含め、すべての種類のフォントを使用できます。

`fontconfig` ライブラリには、GNOME デスクトップシステムで利用できるすべてのフォントのリストがあります。このリストを作成するために、`fontconfig` は `/etc/fonts/fonts.conf` ファイルに一覧表示されるディレクトリを検索します。システムで利用できるすべてのフォントを表示するには、システム上のファイルマネージャ内で `fonts:///` にアクセスします。

`fontconfig` ライブラリについては、次の URL を参照してください。

<http://freedesktop.org/software/fontconfig>

フォント置換

`fontconfig` ライブラリは、フォント全体または個別の文字が存在しない場合にフォントを置換します。存在しないフォントを表示する必要がある場合、`fontconfig` は別の類似フォントの表示を試みます。たとえば、Web ページが Verdana フォントの表示を要求していて、そのフォントがシステムにインストールされていない場合、`fontconfig` は、Helvetica などの類似フォントを表示します。類似フォントのリストは、`/etc/fonts/fonts.conf` ファイルに定義されています。

選択したフォントには存在しない文字を表示する必要がある場合、fontconfig は別の類似フォントの文字の表示を試みます。たとえば、テキストエディタ アプリケーションのフォントとして Bitstream Vera Sans を選択したとします。Bitstream Vera フォントファミリにはキリル文字が含まれません。キリル文字が含まれた文書を開くと、テキストエディタ はキリル文字を含む類似フォントを使用して文字を表示します。

fontconfig ライブラリは、serif、sans-serif、および monospace など、フォントの別名も定義します。フォントの別名の 1 つを選択すると、`/etc/fonts/fonts.conf` 内でその別名に定義されている最初のフォントが使用されます。

すべてのユーザーにフォントを追加する

すべてのユーザーにフォントを追加するには、次の手順を実行します。

1. `/etc/fonts/fonts.conf` ファイル内のディレクトリの 1 つに、フォントファイルをコピーします。通常、フォントは `/usr/share/fonts/` ディレクトリに格納されています。
2. fontconfig ライブラリは、フォントのリストを自動的に更新します。フォントのリストが更新されない場合は、次のコマンドを実行します。

```
# fc-cache
```

また、ディレクトリを特定して更新するには、次のコマンドを実行します。

```
# fc-cache directory-name
```

また、`/etc/fonts/fonts.conf` にあるディレクトリのみから更新し、更新済みディレクトリのリストを表示するには、次のコマンドを実行します。

```
# fc-cache --system-only -v
```

個別ユーザーにフォントを追加する

個別のユーザーにフォントを追加するには、次の手順を実行します。

1. ユーザーの `$HOME/.fonts` ディレクトリに、フォントファイルをコピーします。ファイルマネージャ内の `fonts:///` の場所にフォントファイルをドラッグすると、フォントファイルが `$HOME/.fonts` ディレクトリにコピーされます。
2. fontconfig ライブラリは、フォントのリストを自動的に更新します。フォントのリストが更新されない場合は、次のコマンドを実行します。

```
# fc-cache directory-name
```

fc-cache コマンドで使用する他のオプションについては、前のセクションを参照してください。

第 5 章

MIME タイプ

この章では、アプリケーションがどのように MIME タイプを検出するかを説明します。また、MIME タイプの登録方法、および GNOME デスクトップへのアプリケーションの追加方法についても説明します。

MIME タイプの概要

MIME (*Multipurpose Internet Mail Extension*) タイプは、ファイルの書式を特定します。アプリケーションは、MIME タイプによってファイルを読み取ることができます。インターネットブラウザや電子メールなどのアプリケーションは、MIME タイプを使用して異なるタイプのファイルを扱います。たとえば、電子メールアプリケーションの場合、MIME タイプを使用して電子メールに添付されているファイルのタイプを検知できます。

Nautilus ファイルマネージャは MIME タイプを使用してファイルの種類を識別します。ファイルマネージャは、以下のタスクを実行するために MIME タイプを知る必要があります。

- 正しいアプリケーションでファイルを開く
- ファイルのタイプを説明する文字列を表示する
- ファイルを表す適切なアイコンを表示する
- そのファイルを開くことが可能な別のアプリケーションのリストを表示する

新しいアプリケーションを追加する場合は、そのアプリケーションに関連付けられるファイルを、ほかのアプリケーションが認識できる必要があります。ほかのアプリケーションがそのアプリケーションファイルの MIME タイプを検出するためには、いくつかのタスクを実行する必要があります。

この節では、アプリケーションはどのようにファイルの MIME タイプを検出するか、またアプリケーションはどのように MIME タイプに関連付けられるのかを説明します。また、この章では、新しいアプリケーションを追加するための手順についても説明します。

ファイルの MIME タイプの検出

アプリケーションは以下の方法でファイルの MIME タイプを検出します。

1. アプリケーションは、「ファイルタイプ判別パターン」を使用して、ファイル内の特定のパターンを検索します。ファイルタイプ判別パターンは、ファイル内の特定のパターンと MIME タイプを関連付けます。アプリケーションがそのパターンとの一致を見つけると、そのパターンに関連付けられている MIME タイプがファイルの MIME タイプになります。
2. ファイルタイプ判別パターンによって MIME タイプが識別されない場合、アプリケーションはファイル名を調べます。アプリケーションは、ファイル名を「MIME タイプレジストリ」と照合します。MIME タイプレジストリは、特定のファイル拡張子とファイル名パターンを特定の MIME タイプに関連付けます。ファイル名との一致が見つかり、その拡張子またはパターンに関連付けられている MIME タイプが、ファイルの MIME タイプになります。

この節では、ファイルタイプ判別パターンと MIME タイプレジストリについて詳しく説明します。

ファイルタイプ判別パターン

ファイルタイプ判別パターンは、`/etc/gnome-vfs-mime-magic` ファイルに指定されています。以下にファイルタイプ判別パターンの例を示します。

```
0 string \x89PNG image/png
```

ファイルタイプ判別パターンの構文は、以下のとおりです。

`offset_start[:offset_end] pattern_type pattern [&pattern_mask] type`

表 5-1 は、ファイルタイプ判別パターンの各フィールドを説明しています。

表 5-1 ファイルタイプ判別パターンのフィールド

フィールド	説明
<code>offset_start</code>	テキストパターンを判別する前に、ファイル内で無視する文字数を指定します。
<code>pattern_type</code>	判別するパターンのタイプを指定します。このマニュアルの発行時点では、 <code>string</code> パターンタイプのみがサポートされています。
<code>pattern</code>	判別するパターンを指定します。

表 5-1 ファイルタイプ判別パターンのフィールド (続き)

フィールド	説明
pattern_mask	パターンマスクは、16 進数形式で指定します。パターンマスクについては、次の節を参照してください。 このフィールドは省略可能です。このフィールドは例には示されていません。
type	このエントリと一致するファイルに関連付ける MIME タイプを指定します。

パターンマスク

パターンマスクは、ファイル内でパターンを判別するときに、パターン中で無視するビットを識別します。以下は、ファイルタイプ判別パターンとパターンマスクの例を示しています。

```
0 string BMxxxx\000\000 &0xffff00000000ffff image/bmp
```

この例のパターンとマスクは、以下のとおりです。

パターン	B	M	x	x	x	x	\000	\000
マスク	ff	ff	00	00	00	00	ff	ff

パターンとマスクは、次のような特性でファイルを指定します。

1. ファイルは BM から始まります。
2. BM の後に、任意の 4 バイトの値が続きます。
3. 任意の 4 バイトの後に、\000\000 が続きます。

ファイルタイプ判別パターンは、このパターンとマスクに一致するファイルの MIME タイプが、image/bmpであることを示します。

MIME タイプレジストリ

MIME タイプレジストリは、/usr/share/mime-info にあります。MIME タイプレジストリには、次のファイルが含まれています。

ファイル	ファイル拡張子
MIME 情報ファイル	.mime
MIME キーファイル	.keys

この節では、MIME 情報ファイルと MIME キーファイルについて説明します。

MIME 情報ファイル

MIME 情報ファイルは、MIME タイプを以下の 1 つまたは両方と関連付けます。

- ファイル拡張子
- ファイル名のパターン

アプリケーションがファイルの MIME タイプを検索する場合、アプリケーションはファイル名を MIME 情報ファイルと照合します。ファイル名との一致が見つかったら、その拡張子またはパターンに関連付けられている MIME タイプが、ファイルの MIME タイプになります。

MIME 情報ファイルでは、検索するファイル名のパターンは、正規表現で書かれています。

MIME 情報ファイル内の MIME タイプエントリの形式は、以下のとおりです。

MIME-type
ext[,priority]: list-of-extensions
regex[,priority]: list-of-regular-expressions

ファイル拡張子および正規表現に、優先順位値を指定できます。優先順位値を使用して、複合されたファイル名を区別できます。たとえば、.gz 拡張子に優先順位値 1 を割り当て、それより高い優先順位値の 2 を .tar.gz 拡張子に割り当てたとしてします。この場合、ファイル abc.tar.gz の MIME タイプは、.tar.gz になります。

注 – ext フィールドと regex フィールドは、タブ文字 (\t) を使用してインデントする必要があります。

以下の MIME タイプエントリは、gnome-vfs.mime MIME 情報ファイルからのサンプルです。

```
application/x-compressed-tar
  regex,2: tar\.gz$
  ext: tgz
audio/x-real-audio
  ext: rm ra ram
image/jpeg
  ext: jpe jpeg jpg
image/png
  ext: png
text/html
  ext: html htm HTML
text/plain
  ext: asc txt TXT
text/x-readme
  regex: README.*
```

注 – ファイルマネージャは、MIME 情報ファイルをアルファベット順に読み取ります。MIME タイプがファイル拡張子または正規表現と関連付けられる順序は、アルファベット順で決定します。たとえば、ファイル abc.mime と def.mime で、同じファイル拡張子が異なる MIME タイプに割り当てられている場合、abc.mime にある MIME タイプが使用されます。

MIME キーファイル

MIME キーファイルは、ユーザーインタフェースで使用する MIME タイプに関する情報を提供します。たとえば、MIME キーファイルは、MIME タイプの説明を提供し、その MIME タイプのファイルを表すアイコンを指定します。

以下の例は、MIME キーファイルのサンプルを示しています。

```
text/html
  description=HTML page
  icon_filename=gnome-text-html
  default_action_type=application
  short_list_application_ids_for_novice_user_level=mozilla,netscape,galeon
  category=Documents/World Wide Web
```

注 – MIME キーファイル内のキーは、タブ文字 (\t) を使用してインデントする必要があります。

表 5-2 は、MIME キーファイルで最も重要なキーについて説明しています。通常、description キーと category キーは使用しているシステム環境に対応します。

表 5-2 MIME キーファイル内のキー

キー	説明
can_be_executable	この MIME タイプのファイルが実行可能かどうかを指定します
description	MIME タイプの説明。この説明は、ファイルマネージャおよびほかのアプリケーションで表示できます
icon_filename	MIME タイプを表すアイコンのファイル名を指定します。ファイル名へのパス、あるいはファイル拡張子は指定しない このアイコンは、ファイルマネージャおよびほかのアプリケーションで表示できます

表 5-2 MIME キーファイル内のキー (続き)

キー	説明
default_action_type	この MIME タイプのファイルを開いたときに 行われる動作のカテゴリを指定します。ほと んどのアプリケーションのこの MIME タイプ に、application を指定します
short_list_application_ids _for_novice_user_level	この MIME タイプのファイルを開くときに使 用するアプリケーションを指定します。優先 順に 1 つ以上のアプリケーションを指定しま す。アプリケーションは、アプリケーション レジストリに登録する必要があります
category	MIME タイプのカテゴリを指定します。この キーの値によって、「ファイルの関連付け」 設定ツール内の MIME タイプの場所が決まり ます。

MIME タイプのためのアプリケーション の登録

アプリケーションレジストリには、アプリケーションを登録するテキストファイルが含まれています。このアプリケーション登録ファイルには、アプリケーションの詳細を指定する一連のキーと値の組み合わせが含まれています。アプリケーション登録ファイルには、以下の情報が含まれています。

- アプリケーションの起動に使用するコマンド
- アプリケーションに関連付ける MIME タイプ

アプリケーション登録ファイルには、1 つ以上のアプリケーションを登録することができます。アプリケーション登録ファイルには、.applications 拡張子が付きます。

アプリケーションレジストリの場所は、/usr/share/application-registry です。このディレクトリには、デフォルトのアプリケーション登録ファイル (gnome-vfs.applications) があります。

アプリケーションを登録するには、アプリケーションレジストリにそのアプリケーションの登録ファイルを追加します。

以下は、アプリケーション登録の例です。

```
eog
  command=eog
  name=Eye of Gnome
  can_open_multiple_files=true
```

```

expects_uris=false
requires_terminal=false
mime_types=image/bmp,image/gif,image/jpeg,image/png,image/tiff,
image/x-xpixmap,image/x-bmp,image/x-png,image/x-portable-anymap,
image/x-portable-bitmap,image/x-portable-graymap,
image/x-portable-pixmap

```

表 5-3 は、アプリケーション登録ファイル内のキーを説明しています。

表 5-3 アプリケーション登録のためのキー

キー	説明
アプリケーション識別子	アプリケーションの一意の識別子を指定します。この識別子は、このアプリケーション用の MIME キーファイル内の <code>short_list_application_ids_for_novice_user_level</code> キー内の識別子と同じにする必要があります
<code>command</code>	アプリケーションの起動に使用するコマンド、およびコマンドと共に使用するオプションを指定します
<code>name</code>	アプリケーションの名前を指定します。この名前は、ユーザーインターフェースに表示されます。たとえば、この名前は、ファイルマネージャの「アプリケーションから開く」サブメニューに表示されます
<code>can_open_multiple_files</code>	アプリケーションが複数のファイルを同時に開くことができるかどうかを指定します
<code>expects_uris</code>	アプリケーションが URI を処理できるかどうかを指定します。このキーの値が <code>true</code> の場合、アプリケーション登録エントリ内に <code>supported_uri_schemes</code> キーが必要です
<code>supported_uri_schemes</code>	アプリケーションが処理できる URI スキーマを指定します
<code>requires_terminal</code>	アプリケーションを端末で実行するかどうかを指定します。アプリケーションが実行に使用するウィンドウを作成しない場合は、このフィールドに <code>true</code> を入力します
<code>mime_types</code>	アプリケーションが使用できる MIME タイプを指定します

GNOME デスクトップへのアプリケーションの追加

GNOME デスクトップにアプリケーションを追加するには、次の手順を実行します。

1. アプリケーション用にメニュー項目を追加します。メニュー項目の追加方法については、[第 2 章](#)を参照してください。
2. `/usr/share/icons/theme-name/icon-size/apps` に、アプリケーション用のアイコンを追加します。アイコンおよびテーマの詳細については、[第 3 章](#)を参照してください。
3. アプリケーションが新しい MIME タイプを使用する場合は、その新しい MIME タイプ用のファイルタイプ検索ツールを追加します。ファイルタイプ検索ツールの詳細については、[52 ページの「ファイルタイプ判別パターン」](#)を参照してください。
4. アプリケーションが新しい MIME タイプを使用する場合は、そのアプリケーション用の MIME 情報ファイルを MIME タイプレジストリに追加します。MIME 情報ファイルの詳細については、[54 ページの「MIME 情報ファイル」](#)を参照してください。
5. アプリケーション用の MIME キーファイルを MIME タイプレジストリに追加します。MIME キーファイルの詳細については、[55 ページの「MIME キーファイル」](#)を参照してください。
6. アプリケーションが新しい MIME タイプを使用する場合は、その MIME タイプ用のアイコンを `/usr/share/icons/theme-name/icon-size/mimetypes` に追加します。アイコンおよびテーマの詳細については、[第 3 章](#)を参照してください。
7. アプリケーションと MIME タイプを関連付けるには、アプリケーション登録ファイルをアプリケーションレジストリに追加します。アプリケーションレジストリの詳細については、[56 ページの「MIME タイプのためのアプリケーションの登録」](#)を参照してください。

第 6 章

スクリーンセーバーの設定

この章では、スクリーンセーバーの設定を行う方法について説明します。この章では、スクリーンセーバーに利用可能なディスプレイの変更方法についても説明します。

スクリーンセーバーの概要

スクリーンセーバーは、画面が使用されていないときに、画面上のイメージを置き換えるアプリケーションです。Java Desktop System 用のスクリーンセーバーアプリケーションは、XScreenSaver です。この節では、XScreenSaver アプリケーションの設定方法、スクリーンセーバーに利用可能なディスプレイの変更方法について説明します。

スクリーンセーバーの設定

デフォルトのスクリーンセーバー設定は、XScreenSaver ファイルに格納されています。このファイルの格納場所は、次の表に示すように、オペレーティング環境によって異なります。

オペレーティング環境	ファイルの場所
Linux	/usr/X11R6/lib/X11/app-defaults/XScreenSaver
Solaris	/usr/openwin/lib/app-defaults/XScreenSaver

スクリーンセーバーのアプリケーション設定を変更するには、スクリーンセーバー設定ツールを使用します。スクリーンセーバー設定を変更すると、ホームディレクトリに `$HOME/.xscreensaver` ファイルとして格納されます。スクリーンセーバーの設定については、最新バージョンの『*Java Desktop System Release 2 ユーザーズガイド*』を参照してください。

コマンドを実行してスクリーンセーバー設定ツールを起動することもできます。使用するコマンドは、以下の表に示すように、オペレーティング環境によって異なります。

オペレーティング環境	コマンド
Linux	<code>/usr/bin/xscreensaver-demo</code>
Solaris	<code>/usr/openwin/bin/xscreensaver-demo</code>

すべてのユーザー用にデフォルトのスクリーンセーバー設定を行うには、XScreensaver ファイルを変更します。また、「**XScreensaver**」ダイアログを使用して、`$HOME/.xscreensaver` ファイルを作成し、そのファイルを XScreensaver ファイルの場所にコピーすることもできます。

ユーザーのデフォルト設定を復元する場合は、そのユーザーのホームディレクトリから `$HOME/.xscreensaver` ファイルを削除します。`$HOME/.xscreensaver` ファイルが存在しない場合は、XScreensaver ファイルのデフォルト設定が使用されています。

注 – XScreensaver のデフォルトの表示動作では、ブランク画面が表示されます。ブランク画面ではユーザーが困惑する恐れがあります。したがって、デフォルトの表示動作を変更することをお勧めします。

スクリーンセーバー設定の変更を有効にするには、以下のコマンドを使用してスクリーンセーバー設定を再読み込みします。

```
# xscreensaver-command -restart
```

注 – このマニュアルの発行時点では、xset アプリケーションは、XScreensaver で機能していません。スクリーンセーバー設定を変更するには、XScreensaver ファイルまたは `$HOME/.xscreensaver` ファイルを変更します。

スクリーンセーバーディスプレイの変更

スクリーンセーバーアプリケーションでは、1つ以上のスクリーンセーバーディスプレイを選択できます。スクリーンセーバーディスプレイは、画面が使用されていないときに、画面上にイメージを表示するアプリケーションです。スクリーンセーバーディスプレイは、XScreenSaver ファイルと `$HOME/.xscreensaver` ファイルに記述されています。

Linux 上でスクリーンセーバーディスプレイを追加する

新しいスクリーンセーバーディスプレイを追加するには、そのディスプレイの実行ファイルをディスプレイ用のディレクトリにコピーします。たとえば、Java Desktop System では、ディスプレイは `/usr/lib/xscreensaver` ディレクトリにあります。スクリーンセーバーディスプレイのコマンドを、XScreenSaver ファイルまたは `$HOME/.xscreensaver` ファイルに追加します。ウィンドウ内ではなく、全画面上でスクリーンセーバーディスプレイを実行するために必要となる引数を含めます。たとえば、スクリーンセーバーディスプレイを全画面に表示するために、`-root` オプションを含める場合があります。

Solaris プラットフォーム上でスクリーンセーバーディスプレイを追加する

新しいスクリーンセーバーディスプレイを追加するには、そのディスプレイ用の実行ファイルを `/usr/openwin/lib/xscreensaver/hacks` ディレクトリにコピーします。スクリーンセーバーディスプレイのコマンドを、XScreenSaver ファイルまたは `$HOME/.xscreensaver` ファイルに追加します。ウィンドウ内ではなく、全画面上でスクリーンセーバーディスプレイを実行するために必要となる引数を含めます。たとえば、スクリーンセーバーディスプレイを全画面に表示するために、`-root` オプションを含める場合があります。

Linux 上でスクリーンセーバーディスプレイを無効化する

スクリーンセーバーディスプレイを無効にするには、設定ファイルで、スクリーンセーバーディスプレイのコマンドの先頭にマイナス記号 (-) を追加します。以下の `$HOME/.xscreensaver` ファイルからの引用は、Qix (solid) スクリーンセーバーディスプレイを無効化した状態を表しています。

- "Qix (solid)" qix -root -solid -segments 100

Solaris プラットフォーム上でスクリーンセーバーディスプレイを無効化する

Sun Ray™ Thin クライアント上のすべてのユーザーに対してスクリーンセーバーディスプレイを無効にするには、`/usr/openwin/lib/xscreensaver/hacks` ディレクトリを `hacks` 以外の名前に変更します。

GNOME デスクトップでは、OpenGL スクリーンセーバーディスプレイが提供されます。デフォルトでは、すべての OpenGL スクリーンセーバーディスプレイが無効になっています。これは、OpenGL ライブラリをインストールしていないユーザーが多いためです。OpenGL をインストール済みのユーザーは、「**XScreenSaver**」ダイアログを使用して、OpenGL スクリーンセーバーディスプレイを有効にできます。

一部のスクリーンセーバーディスプレイでは、ユーザーの画面の内容が表示されます。セキュリティ違反を避けるために、画面の内容を表示するスクリーンセーバーディスプレイは、Sun Microsystems が配布する GNOME デスクトップには含まれません。

注 - XScreenSaver アプリケーション用 プラグイン可能認証モジュール (PAM) のサービス名は、`dtssession` です。この名前は古いアプリケーションと互換性を持たせるために使用しています。

第 7 章

セッション管理

この章では、セッション管理の概要と、セッションのデフォルトを設定する方法について説明します。また、セッションとログインスクリプトに関する情報も提供します。

セッションの概要

デスクトップにログインしてから、ログアウトするまでの間をセッションといいます。ログインマネージャによってユーザーが認証されると、セッションマネージャが開始されます。ユーザーは、セッションマネージャによりセッションを管理できます。たとえば、セッションの状態を保存して、次回ログインしたときにそのセッションに戻ることができます。

少なくとも、以下のアプリケーションがセッションで動作します。

- `gnome-session` セッションマネージャ
- GConf X 設定デーモンの `gnome-settings-daemon`
- `gnome-panel` アプリケーション (デスクトップ上でパネルを実行する)
- Metacity ウィンドウマネージャ
- Nautilus ファイルマネージャ
- `gnome-keyring` デーモン
- CD デバイスオプション設定ツール (`magicdev`) がシステムに存在する場合、この設定ツールもセッション内で実行されます。

セッションのデフォルトの設定

以下の表は、デフォルトのセッション情報が含まれているファイルを示しています。

ファイル	説明
<code>/usr/share/gnome/default.session</code>	デフォルトのセッションファイル。デフォルトのセッションの詳細は、このファイルに格納されています
<code>\$HOME/.gnome2/session</code>	ユーザーセッションファイル。セッションを変更した場合、その詳細がこのファイルに格納されます

すべてのユーザー用にデフォルトのセッションの詳細を設定する場合は、デフォルトのセッションファイルを変更します。

ユーザーのデフォルトのセッション設定を復元する場合は、そのユーザーのホームディレクトリからセッションファイルを削除します。ユーザーセッションファイルが存在しない場合は、`/usr/share/gnome/default.session` のデフォルト設定が使用されています。

現在のセッションをデフォルトセッションとして保存する場合、`gnome-session-save` コマンドを実行します。

Solaris プラットフォーム上でのログインスクリプトの使用

GNOME デスクトップは、システム全体のログインスクリプトまたはログアウトスクリプトをサポートしていません。セッションマネージャでは、ユーザーセッションの設定だけが可能です。ユーザーは、セッション設定ツールを使用して、非セッション管理のスタートアップアプリケーションを設定できます。

ヒント – システム全体のログインスクリプトのサポートがなくても、作業は可能です。システム全体のログインスクリプトを作成するには、`/usr/dt/config/Xsession.d` ディレクトリにスクリプトを作成します。次のように、スクリプトに名前を付けます。

number-greater-than-1000.name

たとえば、「1050.login」のような名前を付けます。
`/usr/dt/config/Xsession.d` ファイルについては、*dtlogin* マニュアルページの「*The XSession File*」を参照してください。

第 8 章

ヘルプシステム

この章では、GNOME デスクトップのヘルプシステムについて説明します。

ヘルプシステムの概要

GNOME デスクトップは、Yelp ヘルプブラウザにヘルプを表示します。ヘルプのソース文書は XML ファイルです。XML ファイルは、DocBook XML Version 4.1.2 DTD (Document Type Definition) に書き込まれます。DocBook XML ファイルは HTML に変換され、この HTML がヘルプブラウザに表示されます。DocBook XML については、以下の URL を参照してください。

<http://www.oasis-open.org/docbook/xml>

ヘルプシステムは、OMF (*Open Source Metadata Framework*) およびカタログ作成システムを使用します。次の節では、OMF ファイルとカタログ作成システムについて説明します。

OMF ファイル

各マニュアルの XML ファイルには、OMF ファイルが関連付けられています。OMF ファイルには、ヘルプブラウザが使用するマニュアルについての情報が含まれています。OMF ファイルの拡張子は、`.omf` です。

アプリケーションをインストールすると、OMF ファイルが OMF ファイル用のディレクトリにインストールされます。OMF ファイルディレクトリ内にヘルプドキュメントの関連 OMF ファイルがある場合は、ヘルプドキュメントがヘルプブラウザに表示されます。

OMF ファイルには、ヘルプドキュメントに関する以下のような情報が含まれています。

- ヘルプドキュメントの XML ファイルがある場所
- ヘルプドキュメントの題名
- ヘルプドキュメントが属しているサブジェクトカテゴリ

ScrollKeeper カタログ作成システム

ScrollKeeper は、ドキュメント用カタログシステムです。ヘルプブラウザは、ScrollKeeper を使用してシステム上にマニュアルのカタログを作成します。ScrollKeeper は、ヘルプブラウザのために OMF ファイル 内の情報を管理します。

ScrollKeeper が OMF ファイル内の情報を使用することで、ヘルプブラウザはシステム上のヘルプドキュメントを検索できます。アプリケーションをインストールすると、OMF ファイルが OMF ファイル用のディレクトリにインストールされます。

ScrollKeeper は、OMF ファイルが関連付けられた各ヘルプドキュメントの目次を作成します。ScrollKeeper は、OMF ファイル内に指定された XML ファイルから目次を作成します。ヘルプドキュメントの目次はヘルプブラウザウィンドウの左区画に表示されます。ユーザーは、目次を使って左区画に表示されたヘルプドキュメントに移動できます。OMF ファイルのドキュメントの題名もヘルプブラウザウィンドウの左区画に表示されます。

ScrollKeeper には、ヘルプドキュメントが属しているサブジェクトカテゴリの階層が含まれています。ヘルプドキュメントは、ヘルプブラウザウィンドウの左区画のサブジェクトカテゴリ階層に整理されています。OMF ファイル内の文書のサブジェクトカテゴリは、階層内の文書の位置を決定します。サブジェクトカテゴリの階層は、以下のファイルで定義されます。

```
/usr/share/scrollkeeper/Templates/C/scrollkeeper_cl.xml
```

文書のサブジェクトカテゴリは、以下のファイルのサブジェクトカテゴリに対応していません。

```
/usr/share/scrollkeeper/Templates/C/scrollkeeper_cl.xml
```

注 – ヘルプドキュメントの XML ファイルを新しい場所に移動した場合、OMF ファイル内の場所情報も更新しなければならない場合があります。

第 9 章

パフォーマンスの改善

この章では、Java Desktop System のパフォーマンスを改善する方法について説明します。

パフォーマンスの改善の概要

この章では、Java Desktop System のパフォーマンスを改善するために変更できるいくつかの設定について説明します。ユーザー設定について値を設定するには、`gconftool-2` コマンドを使用します。この章のコマンド例では、ユーザー設定ソースの値を変更する方法を示します。

`--direct` および `--config-source` オプションを使用して、設定の必須値またはデフォルト値をセットできます。また、スクリプト内で `gconftool-2` コマンドを使用して、多数の設定値をセットすることもできます。`gconftool-2` コマンドと、このコマンドに使用できるオプションについては、[第 1 章](#)を参照してください。

CPU 使用率の低減

この節では、Java Desktop System の CPU 使用率を減らすための設定について説明します。

CPU リソースをあまり必要としないコントロール テーマオプションを使用する

一部のコントロールテーマオプションは、CPU リソースが限られたシステムでは動作が遅くなる恐れがあります。CPU 使用率を減らすには、以下のいずれかのコントロールオプションを使用します。

- デフォルト値
- Mist
- Simple
- ThinIce

コントロールテーマオプションを変更するには、以下のコマンドを実行します。

```
# gconftool-2 --type string --set  
/desktop/gnome/interface/gtk_theme option-name
```

Simple を使用する場合は、次のコマンドを実行します。

```
# gconftool-2 --type string --set  
/desktop/gnome/interface/gtk_theme Simple
```

または、テーマ設定ツールで適当なオプションを選択します。

CPU リソースをあまり必要としないウィンドウフ レームオプションを使用する

一部のウィンドウフレームテーマオプションは、ウィンドウフレームを描くのにイメージファイルを読み込みます。ほかのオプションは、より簡単な方法でウィンドウフレームを描きます。

Crux ウィンドウフレームオプションは画像ファイルをロードするので、CPU リソースが限られたシステムでは動作が遅くなる恐れがあります。CPU 使用率を減らすには、以下のいずれかのウィンドウフレームオプションを使用します。

- Atlanta
- Esco
- Mist
- Simple

以下のウィンドウフレームオプションも、Crux よりは CPU リソースの使用率が少なくなります。

- AgingGorilla
- Bright
- Metabox

注 – Metabox は、HighContrastInverse などの反転制御オプションとは正しく動作しません。反転制御オプションを使用する場合は、Atlanta を使ってください。

ウィンドウフレームテーマのオプションを変更するには、以下のコマンドを実行します。

```
# gconftool-2 --type string --set /apps/metacity/general/theme  
option-name
```

Atlanta を使用する場合は、次のコマンドを実行します。

```
# gconftool-2 --type string --set /apps/metacity/general/theme  
Atlanta
```

または、テーマ設定ツールで適当なオプションを選択します。

ヒント – Metacity テーマビューアを使用して、ウィンドウフレームオプションのパフォーマンスを測定したり、オプションをプレビューできます。Metacity テーマビューアを起動するには、次のコマンドを使用します。

```
# metacity-theme-viewer option-name
```

たとえば、Atlanta のパフォーマンスを測定し、Atlanta をプレビューするには、次のコマンドを実行します。

```
# metacity-theme-viewer Atlanta
```

メニュー上のアイコンをオフにして CPU 使用率を減らす

メニュー上のいくつかの項目は、その項目の横にアイコンが表示されます。この機能をオフにするには、次のコマンドを実行します。

```
# gconftool-2 --type bool --set  
/desktop/gnome/interface/menus_have_icons false
```

または、Menus & Toolbars 設定ツールで「メニューにアイコンを表示」オプションの選択を解除します。

スプラッシュ画面をオフにして CPU 使用率を減らす

デフォルトでは、デスクトップ環境にログインすると、スプラッシュ画面が表示されます。ログインしている間は、スプラッシュ画面にアイコンが表示されています。スプラッシュ画面をオフにすると、ログインしている間の CPU 使用率を減らせます。

スプラッシュ画面をオフにするには、次のコマンドを実行します。

```
# gconftool-2 --type bool --set  
/apps/gnome-session/options/show_splash_screen false
```

または、セッション 設定ツールで「ログイン時にスプラッシュ画面を表示する」オプションの選択を解除します。

パネルアニメーションをオフにして CPU 使用率を減らす

パネルを表示または非表示にするときに、動画形式でパネルを表示または非表示にすることができます。パネルアニメーションをオフにするには、次のコマンドを実行します。

```
# gconftool-2 --type bool --set  
/apps/panel/global/enable_animations false
```

または、パネル 設定ツールで「引出しとパネルのアニメーション」オプションの選択を解除します。

ファイルマネージャのパフォーマンスの改善

Nautilus ファイルマネージャの一部の機能を変更して、パフォーマンスを改善できます。

パフォーマンス設定を変更する

ファイルマネージャには、パフォーマンスに関連する設定があります。各パフォーマンス設定は、3 つの値のうち 1 つを持ちます。以下の表は、3 つの値を説明しています。

値	説明
always	ローカルファイルと、別のファイルシステム上のファイルの両方に対して、動作を実行する
local_only	ローカルファイルに対してのみ動作を実行する。パフォーマンス設定を local_only セットすると、CPU 使用率は減少する
never	動作を実行しない。パフォーマンス設定を never にセットすると、CPU 使用率とネットワークトラフィックは減少する

以下の表は、ファイルマネージャのパフォーマンス設定を説明しています。パフォーマンスを最大にするには、設定値を `never` に設定します。

設定	説明
show_icon_text	ファイルを表すアイコンで、テキストファイルの内容をいつプレビューするかを指定します。テキストファイルの内容をプレビューしない場合は、次のコマンドを実行します <pre># gconftool-2 --type string --set /apps/nautilus/preferences/show_icon_text never</pre> または、次の手順を実行します 1. ファイルマネージャウィンドウで「編集」→「設定」を選択して、「プレビュー」を選択します。 2. 「アイコンにテキストを表示」設定のオプションを選択します
show_directory_item_counts	フォルダ内のアイテム数をいつ表示するかを指定します。フォルダ内の項目数を表示しない場合は、次のコマンドを実行します。 <pre># gconftool-2 --type string --set /apps/nautilus/preferences/show_directory_item_counts never</pre> または、次の手順を実行します 1. ファイルマネージャウィンドウで「編集」→「設定」を選択して、「プレビュー」を選択します。 2. 「アイテム数のカウント」設定のオプションを選択します。

設定	説明
show_image_thumbnails	イメージファイルのサムネイルをいつ表示するかを指定します。サムネイルを表示しない場合は、次のコマンドを実行します <pre># gconftool-2 --type string --set /apps/nautilus/preferences/show_image_thumbnails never</pre> または、次の手順を実行します 1. ファイルマネージャウィンドウで「編集」→「設定」を選択して、「プレビュー」を選択します。 2. 「サムネイルの表示」設定のオプションを選択します
preview_sound	サウンドファイルの内容をいつプレビューするかを指定します。サウンドファイルの内容をプレビューしない場合は、次のコマンドを実行します <pre># gconftool-2 --type string --set /apps/nautilus/preferences/preview_sound never</pre> または、次の手順を実行します 1. ファイルマネージャウィンドウで「編集」→「設定」を選択して、「プレビュー」を選択します。 2. 「サウンドファイルをプレビュー」設定のオプションを選択します

サイド区画、ツールバー、ロケーションバーをオフにする

ファイルマネージャには、サイド区画とツールバーをオフにできる設定が含まれています。サイド区画とツールバーをオフにすると、ファイルマネージャのパフォーマンスが改善します。

サイド区画をオフにするには、次のコマンドを実行します。

```
# gconftool-2 --type bool --set /apps/nautilus/preferences/start_with_sidebar false
```

ツールバーをオフにするには、次のコマンドを実行します。

```
# gconftool-2 --type bool --set /apps/nautilus/preferences/start_with_toolbar false
```

ロケーションバーもオフに設定できます。Ctrl + L キーボードショートカットを使用して、ロケーションバーを必要に応じて表示できます。

ロケーションバーをオフにするには、次のコマンドを実行します。

```
# gconftool-2 --type bool --set  
/apps/nautilus/preferences/start_with_location_bar false
```

デスクトップの背景をオフにする

ファイルマネージャには、Nautilus を使用してデスクトップの背景を管理できる設定が含まれています。デスクトップの背景を無効にすることで、パフォーマンスを改善できます。デスクトップの背景を無効にした場合、次の操作を行えません。

- 「デスクトップの背景」メニューの使用
- ファイルマネージャを使用したデスクトップ背景のパターンまたは色の変更
- 「ごみ箱」などのデスクトップの背景オブジェクトの使用。デスクトップの背景オブジェクトは、デスクトップの背景に表示されません。

デスクトップの背景を無効にするには、次のコマンドを実行します。

```
# gconftool-2 --type bool --set  
/apps/nautilus/preferences/show_desktop false
```

省リソースモードを有効化して CPU 使用率を減らす

Metacity ウィンドウマネージャは、ウィンドウの移動およびサイズ変更の際にウィヤーフレームとしてウィンドウを描画するための、省リソースモードを備えています。省リソースモードが有効な場合は、ウィンドウを移動したりサイズ変更したりするとき、ウィンドウのアウトラインだけが表示されます。ウィンドウのコンテンツは、移動中やサイズの変更中に更新される必要はありません。ウィンドウのコンテンツは、移動やサイズ変更の完了時に表示されます。また、省リソースモードが有効な場合は、ウィンドウの移動およびサイズ変更時にウィンドウがアニメーション化されません。

省リソースモードを有効にするには、以下のコマンドを実行します。

```
# gconftool-2 --type bool --set  
/apps/metacity/general/reduced_resources true
```

Solaris プラットフォームで mediaLib ライブラリを使用する

mediaLib™ ライブラリは、共通のマルチメディア操作を高速化するための関数のコレクションです。ライブラリがシステムにインストールされている場合、Java Desktop System では mediaLib ライブラリが使用されます。Java Desktop System のパフォーマンス、特に Nautilus は、mediaLib ライブラリがインストールされていると著しく向上します。Java Desktop System では、バージョン 2.0 以降の mediaLib が必要です。

mediaLib ライブラリの入手方法およびインストール方法については、mediaLib の Web サイト (<http://www.sun.com/processors/vis/mlib.html>) を参照してください。

X ウィンドウシステムのネットワークトラフィックの低減

この節では、Java Desktop System で X ウィンドウシステム のネットワークトラフィックを減少させる設定について説明します。

ネットワークトラフィックの少ないウィンドウフレームテーマオプションを使用する

リモートディスプレイプロトコルは、ピクセルブロック内のピクセルがすべて同じ色の場合、そのブロック内のすべてのピクセルを転送しません。

X ウィンドウシステムのネットワークトラフィックを減らすには、単一色を使用するウィンドウフレームテーマオプションを使用します。つまり、次のいずれかのウィンドウフレームオプションを使用します。

- Mist
- Simple

テーマオプションの変更方法については、70 ページの「[CPU リソースをあまり必要としないウィンドウフレームオプションを使用する](#)」を参照してください。

メニュー上のアイコンをオフにしてネットワークトラフィックを減らす

メニュー上のいくつかの項目は、その項目の横にアイコンが表示されます。アイコンが別のファイルシステム上にある場合、この機能が原因で X ウィンドウシステムのネットワークトラフィックが増大します。また、パネルがリモートホスト上で表示される場合も、X ウィンドウシステムのネットワークトラフィックが増大します。たとえば、Sun Ray サーバがパネルを実行すると、Sun Ray クライアント上にもそのパネルが表示されます。

この機能を無効にする方法については、71 ページの「[メニュー上のアイコンをオフにして CPU 使用率を減らす](#)」を参照してください。

省リソースモードを有効化してネットワークトラフィックを減らす

ウィンドウを表示するアプリケーションがリモートホスト上で実行している場合、省リソースモードを有効化して X Window System のネットワークトラフィックを減らします。

省リソースモードについては、[75 ページの「省リソースモードを有効化して CPU 使用率を減らす」](#)を参照してください。

第 10 章

Java Desktop System の機能の無効化

この章では、Java Desktop System の特定の機能が無効にする方法について説明します。

Java Desktop System の機能の無効化の概要

Java Desktop System には、Java Desktop System の特定の機能へのアクセスを制限する機能があります。無効化機能は、コンピュータ上で実行できるユーザーのアクションを制限したい場合などに役立ちます。たとえば、トレードショーにコンピュータを出展し、誰もが操作できるようにした場合は、コマンドラインでの操作を無効にした方が安全です。無効化機能は、ロックダウン機能としても知られています。

機能が無効化するには、GConf キーを設定します。GConf キーの設定方法については、第 1 章を参照してください。GConf エディタアプリケーションを使用して、設定ソースの GConf キーを設定することもできます。GConf エディタアプリケーションについては、『GConf エディタのマニュアル』を参照してください。

スクリーンロックを無効にする

スクリーンロック機能が無効にするには、
`/apps/panel/global/disable_lock_screen` キーを `true` に設定します。

スクリーンロック機能が無効化すると、次の項目がパネルから削除されます。

- 「メインメニュー」の「画面のロック」メニュー項目

- 「パネルに追加」 → 「アクション」メニューの「ロック」メニュー項目。このメニューを開くには、パネル上の空の領域で右クリックし、「パネルに追加」 → 「アクション」を選択します。
- 「メニューバー」アプレット上の「アクション」メニューの「画面のロック」メニュー項目。

また、パネル上の「画面のロック」ボタンも無効になります。

ログアウトを無効にする

ログアウト機能を無効にするには、`/apps/panel/global/disable_log_out` キーを `true` に設定します。

ログアウト機能を無効化すると、次の項目がパネルから削除されます。

- 「メインメニュー」の「*user* をログアウト」メニュー項目
- 「パネルに追加」 → 「アクション」メニューの「ログアウト」メニュー項目
- 「メニューバー」アプレット上の「アクション」メニューの「*user* をログアウト」メニュー項目

また、パネル上の「ログアウト」ボタンも無効になります。

強制終了操作を無効にする

強制終了機能を無効にするには、`/apps/panel/global/disable_force_quit` キーを `true` に設定します。

強制終了機能を無効にすると、「強制終了」メニュー項目が、「パネルに追加」 → 「アクション」メニューから削除されます。また、パネル上の「強制終了」ボタンも無効になります。

コマンドライン操作を無効にする

コマンドラインからの操作を無効にするには、
`/desktop/gnome/lockdown/disable_command_line` キーを `true` に設定します。

コマンドライン操作を無効にすると、次の項目がパネルから削除されます。

- 次のコマンドを含むメニュー項目

- GNOME 端末コマンド (`/usr/bin/gnome-terminal`)
- `/usr/bin/xterm`
- `/usr/bin/setterm`

項目は、次のメニューから削除されます。

- メインメニュー:
 - 「パネルに追加」 → 「メニューからランチャー」
- メインメニューおよび「パネルに追加」 → 「ユーティリティ」メニューの「コマンドライン」アプレット
- 次のメニューから「アプリケーションの実行」メニュー項目を実行します。
 - メインメニュー:
 - 「パネルに追加」メニューの「アクション」サブメニュー
 - 「メニューバー」アプレットの「アクション」メニュー

パネル設定を無効にする

パネル設定を無効にするには、`/apps/panel/global/locked_down` キーを `true` に設定します。

パネル設定を無効にすると、ユーザーインターフェースが次のように変更されます。

- 次の項目は、「パネル」ポップアップメニューおよび「引き出し」ポップアップメニューから削除されます。
 - 「パネルに追加」
 - 「パネルを削除」
 - 「プロパティ」
 - 「新規パネル」
- 次の項目は、パネルオブジェクトのポップアップメニューから削除されます。
 - 「パネルから削除」
 - 「ロック」

- 「移動」
- 「プロパティ」メニュー項目は、「ランチャー」ポップアップメニューから削除されます。
- 「メインメニュー」ポップアップメニューは無効になります。
- ランチャーのドラッグ機能は無効になります。そのため、ランチャーをパネルに移動したり、パネルから移動することができません。
- パネルのドラッグ機能は無効になります。そのため、パネルを別の場所に移動することはできません。

アプリケーションを制限する

ユーザーが起動できるアプリケーションを制限するには、次のキーを使用します。

- /desktop/gnome/lockdown/restrict_application_launching
- /desktop/gnome/lockdown/allowed_applications

ユーザーが起動できるアプリケーションを制限するには、
/desktop/gnome/lockdown/restrict_application_launching キーを true
に設定します。

起動できるアプリケーションを制限すると、ユーザーインタフェースが次のように変更されます。

- 「アプリケーションの実行」項目が次のメニューから削除されます。
 - メインメニュー
 - 「パネルに追加」メニューの「アクション」サブメニュー
 - 「メニューバー」アプレットの「アクション」メニュー
- /desktop/gnome/lockdown/allowed_applications キーに記載されているアプリケーションと一致しないメニュー項目はすべて、次のメニューから削除されます。
 - メインメニュー
 - 「パネルに追加」メニューの「アクション」サブメニュー
 - 「パネルに追加」メニューの「メニューからランチャー」サブメニュー
 - 「メニューバー」アプレットの「アクション」メニュー
- パネル、引き出し、またはデスクトップ上のランチャーをダブルクリックしたときに、そのアプリケーションが
/desktop/gnome/lockdown/allowed_applications キーに記載されていないと、警告ダイアログが表示されます。アプリケーションは起動されません。
- ファイルを開こうとしたときに、そのファイルの MIME タイプが
/desktop/gnome/lockdown/allowed_applications キーに記載されていないアプリケーションに関連付けられている場合、警告ダイアログが表示されます。アプリケーションは起動されません。

- 特定のコマンドが `/desktop/gnome/lockdown/allowed_applications` キーに記載されていない場合、それに対応するメニュー項目はメニューから削除されます。次の表を参照してください。

コマンド	メニュー項目	メニュー
<code>/usr/bin/gnome-search-tool</code>	ファイルの検索	メインメニュー 「メニューバー」アプレットの「アクション」メニュー 「パネルに追加」→「アクション」
<code>/usr/bin/yelp</code>	ヘルプ	メインメニュー
<code>/usr/bin/gnome-panel-screenshot</code>	「スクリーンショット」	「パネルに追加」→「アクション」

付録 A

隠しディレクトリ

この付録では、GNOME デスクトップがユーザーのホームディレクトリに追加する隠しディレクトリについて説明します。

隠しディレクトリとは、ディレクトリ名の先頭にピリオド (.) が付くディレクトリのことです。表 A-1 は、GNOME デスクトップがユーザーのホームディレクトリに追加する隠しディレクトリを示しています。

表 A-1 ユーザーのホームディレクトリ内の隠しディレクトリ

ディレクトリ	説明
.esd_auth	GNOME サウンドデーモン用の認証クッキーが含まれています。GNOME サウンドデーモンは、ESD (Enlightened Sound Daemon) です
.gconf	ユーザーの GConf 設定ソースが含まれています。ユーザーが環境を設定すると、新しい設定情報がこの場所に追加されます
.gconfd	以下の GConf デーモンの詳細が含まれています ■ 構成情報 ■ IOR (Interoperable Object Reference) に よって参照されるオブジェクトのロック情報 ■ IOR によって参照されるオブジェクトの状態情報
.gnome	GConf リポジトリには格納されないユーザー固有のアプリケーションデータが含まれています。たとえば、このディレクトリには MIME タイプ情報やユーザーのセッション情報が含まれています

表 A-1 ユーザーのホームディレクトリ内の隠しディレクトリ (続き)

ディレクトリ	説明
.gnome2	GConf リポジトリには格納されないユーザー固有のアプリケーションデータが含まれています。たとえば、このディレクトリは以下のデータを含みます ■ キーボードショートカット情報 ■ ウィンドウの場所に関する情報 ■ パネルランチャー用のデスクトップエントリファイル このディレクトリは、ユーザー固有のメニューデータも含みます。ユーザーがメニューを変更した場合、詳細がここに格納されます
.gnome2-private	このディレクトリは使用されていません。このマニュアルの発行時点では、このディレクトリに機能はありません
.metacity	Metacity ウィンドウマネージャ用のセッションデータが含まれています。
.nautilus	ユーザー固有のファイルマネージャデータが含まれています。たとえば、このディレクトリは次のデータを含みます ■ ユーザーが作業するディレクトリ用のメタデータ ■ ユーザーが追加する Nautilus エンブレム ■ Nautilus デスクトップ背景画像
.themes	ユーザーが追加した制御テーマオプション、ウィンドウフレームテーマオプション、アイコンテーマオプションが含まれています。テーマの設定ツールでテーマを追加できます
.thumbnails	イメージサムネイルが含まれています。イメージサムネイルは、ファイルマネージャで使用されます。ファイルマネージャには、サムネイルイメージの生成の停止を選択できる設定があります
.xscreensaver	スクリーンセーバ構成データとスクリーンセーバ設定データが含まれています

SMB プリンタ設定に関するセキュリティ上の注意事項

この付録では、Microsoft Windows ネットワーク共有でプリンタを設定する際のセキュリティ上の注意事項について説明します。

Microsoft Windows ネットワーク共有は、Samba (SMB) 共有としても知られています。SMB 共有でプリンタを設定するときは、印刷キューにユーザー名とパスワードを入力する必要があります。

印刷キューのユーザー名とパスワードは、`/etc/cups/printers.conf` ファイルに暗号化されないテキストとして保存されます。`root` 特権を持つユーザーは、このファイルに対して読み取り専用のアクセス権を保持しています。したがって、`root` 特権を持つユーザーであれば印刷キューのユーザー名とパスワードを読み取ることができます。

セキュリティ違反の可能性を軽減するには、印刷キューのアクセスに必要なユーザー名とパスワードは、印刷キュー専用のものにしてください。これにより、セキュリティ違反の可能性は、印刷キューの不正使用だけにとどまります。

用語集

.desktop ファイル	「デスクトップエントリファイル」を参照
.directory ファイル	「ディレクトリエントリファイル」を参照
.omf ファイル	「OMF ファイル」を参照
アプリケーションレジストリ	アプリケーションレジストリとは、アプリケーションを登録するテキストファイルが含まれる場所を示す。GNOME デスクトップのアプリケーションレジストリの場所は、以下のとおり /usr/share/gnome/application-registry
デスクトップエントリファイル	メニュー内の項目についての情報を提供するデータファイル。デスクトップエントリファイルは、名前、実行するコマンド、アイコンなどの項目の詳細を指定する。デスクトップエントリファイルには、 .desktop ファイル拡張子が付く
ディレクトリエントリファイル	メニューについての情報を提供するデータファイル。ディレクトリエントリファイルは、メニューの名前、メニューのツールチップ、メニューを表すアイコンなどの詳細を指定する。ディレクトリエントリファイルには、 .directory ファイル拡張子が付く
ファイルタイプ判別パターン	ファイルタイプ判別パターンは、ファイル内で判別するパターンを指定する。ファイルタイプ判別パターンは、 MIME タイプとパターンを関連付ける。パターンとの一致が見つかり、そのパターンに関連付けられている MIME タイプがファイルの MIME タイプになる
GConf 設定ソース	GConf リポジトリ内の保存場所。例: xml:readonly:/etc/gconf/gconf.xml.defaults.
GConf 設定キー	アプリケーション設定に対応する GConf リポジトリ内の要素。たとえば、/apps/gnome-session/options/show_splash_screen キーは、セッション設定ツールの「ログイン時にスプラッシュ画面を表示する」オプションに対応する
GConf パスファイル	GConf 設定ソースとソースの検索順序がリストされたファイル
GConf スキーマ	スキーマキーとスキーマオブジェクトの総称

GConf スキーマ定義ファイル	GConf スキーマ定義ファイルには、特定のアプリケーションのキーが含まれており、それらのキーの特性を定義する。GConf スキーマは、スキーマ定義ファイルから生成される。スキーマ定義ファイルには、 <code>.schemas</code> ファイル拡張子が付く
GConf スキーマキー	設定キーのスキーマオブジェクトを格納するキー。たとえば、 <code>/schemas/desktop/gnome/interface/font_name</code> は、 <code>/desktop/gnome/interface/font_name</code> 設定キーのスキーマキーである
GConf スキーマオブジェクト	設定キーに関する情報を含む設定ソース内の要素。スキーマオブジェクトは、設定キーのデフォルト値、設定キーの説明などの情報を持つ
IOR (Interoperable Object Reference)	IOR は、CORBA オブジェクトに対する文字列参照。IOR は、オブジェクトを制御するためにメッセージを送信できるホスト名とポートをエンコードする。IOR は、オブジェクトを識別するオブジェクトキーを持っている
MIME	Multipurpose Internet Mail Extension の略
MIME 情報ファイル	MIME 情報ファイルは、MIME タイプとファイル拡張子およびファイル名パターンを関連付けるテキストファイル。MIME 情報ファイルには、 <code>.mime</code> ファイル拡張子が付く
MIME キーファイル	MIME キーファイルは、ユーザーインタフェースで使用される MIME タイプに関する情報を提供する。たとえば、MIME キーファイルは、その MIME タイプのファイルを表すアイコンを指定する。MIME キーファイルには、 <code>.keys</code> ファイル拡張子が付く
MIME タイプ	MIME タイプは、ファイルの形式を識別する。アプリケーションは、MIME タイプによってファイルを読み取ることができる。たとえば、電子メールアプリケーションは、 <code>image/png</code> MIME タイプを使用して、電子メールに PNG (Portable Networks Graphic) ファイルが添付されていることを認識する
MIME タイプレジストリ	MIME タイプレジストリとは、デスクトップ環境用の MIME タイプを登録するテキストファイルが含まれる場所を示す。GNOME デスクトップの MIME タイプレジストリの場所は、以下のとおり <code>/usr/share/mime-info</code>
OMF ファイル	Open Source Metadata Framework ファイル。OMF ファイルは、マニュアルの XML ファイルに関連付けられる。OMF ファイルには、ヘルプブラウザが使用するマニュアルについての情報が含まれている。OMF ファイルのファイル拡張子は、 <code>.omf</code>
PAM パターンマスク	プラグイン可能認証モジュール (Pluggable Authentication Module)。パターンマスクは、ファイルタイプ検索ツール内の一連の 16 進数文字。パターンマスクは、ファイル内でパターンを検索するときに、パターン中で無視するビットを識別する

スクリーンセーバー	スクリーンセーバーは、画面が使用されていないときに、画面上のイメージを置き換えるアプリケーションのこと。GNOME デスクトップ用のスクリーンセーバーアプリケーションは、XScreenSaver
スクリーンセーバーディスプレイ	スクリーンセーバーディスプレイは、画面が使用されていないときに、画面上にイメージを表示するアプリケーションのこと
UDP	ユーザーデータグラムプロトコル
URI (Uniform Resource Identifier)	URI (Uniform Resource Identifier) は、ファイルシステム内または Web 上の特定の場所を識別する文字列。たとえば、Web ページのアドレスは URI である
vfolder	システム上の物理的な場所に存在する項目の仮想表現。たとえば、vfolder が、いくつかのディレクトリの内容を表す場合がある。メニューにおいて vfolder は、物理的には複数のディレクトリに置かれている項目のメニュー上の表現になる
vfolder 情報ファイル	XML 形式の vfolder の説明ファイル。vfolder 情報ファイルは、メニューの構造を指定する

索引

C

CPU 使用率, 減らす, 69-76

D

.desktop ファイル, 「デスクトップエントリ
ファイル」を参照
.directory ファイル, 「ディレクトリエントリ
ファイル」を参照

F

fontconfig, 「フォント」を参照

G

GConf

HTTP プロキシの設定, 22
概要, 13
キーボードショートカットの設定, 23
コマンドラインツール, 18-21
スキーマ, 16-17
スキーマオブジェクト, 16-17
スキーマキー, 16-17
スキーマ定義ファイル
概要, 17
パネルおよびパネルオブジェクト, 24-26
スプラッシュイメージの設定, 28-29
設定ソース, 15-16
設定値, 21-22

GConf (続き)

デーモン, 18
デフォルト設定値の復元, 29
背景, 設定, 28
パネルオブジェクトの環境の設定, 23-27
パネルの設定の個別に設定, 23-27
フォントの設定, 27-28
リポジトリ, 14-17
ワークスペースの数の設定, 22-23
gconfd, 「GConf デーモン」を参照
gconftool-2, 「GConf コマンドラインツール」
を参照
GTK+ テーマ, 「テーマ、コントロールオブ
ション」を参照

H

HTTP プロキシ, GConf による設定, 22

M

mediaLib ライブラリ, CPU 使用率, 75-76
Metacity テーマ, 「テーマ、ウィンドウフレイ
ムのオプション」を参照
Microsoft Windows ネットワーク共有,
「SMB」を参照
MIME タイプ
MIME キーファイル, 55-56
MIME 情報ファイル, 54-55
MIME タイプレジストリ, 53-56
アプリケーションの追加, 58

MIME タイプ (続き)

アプリケーションの登録, 56-57

概要, 51

検出, 52-56

パターンマスク, 53

ファイルタイプ判別パターン, 52-53

MIME タイプの検出, 52-56

N

Nautilus, 「ファイルマネージャ」を参照

O

OMF ファイル, 67-68

.omf ファイル, 「OMF ファイル」を参照

S

Samba, 「SMB」を参照

ScrollKeeper カタログ作成システム, 68

SMB, プリンタの構成, 87

V

vfolder, 32-34

vfolder 情報ファイル, 33

X

XScreenSaver, 「スクリーンセーバー」を参照

X ウィンドウシステムのネットワークトラフィック、低減, 76-77

あ

アイコンテーマ, 「テーマ、アイコンオプション」を参照

アイコン、メニュー上の, 「メニューアイコン」を参照

アプリケーション
追加, 58

アプリケーション (続き)

レジストリ, 56-57

お

オンラインヘルプ, 「ヘルプ」を参照

か

隠しディレクトリ, 85-86

き

キーボード, GConf によるショートカットの設定, 23

機能の無効化

アプリケーションを制限する, 82-83

概要, 79

強制終了, 80

コマンドライン, 81

スクリーンロック, 79-80

パネル設定, 81-82

ログアウト, 80

し

省リソースモード

CPU 使用率, 75

ネットワークトラフィック, 77

有効化, 75

す

スキーマ

スキーマオブジェクト, 16-17

スキーマキー, 16-17

スキーマ定義ファイル

概要, 17

パネルおよびパネルオブジェクト, 24-26

説明, 16-17

スクリーンセーバー

概要, 59

スクリーンセーバー (続き)
スクリーンセーバーディスプレイの変更, 61-62
設定, 59-60
スプラッシュ画面
CPU 使用率, 72
イメージ、GConf による設定, 28-29
オフにする, 72

せ

セッション
概要, 63
デフォルトの設定, 64
ログインスクリプト, 64-65
設定値
GConf による設定, 21-22
GConf によるデフォルトへの復元, 29

て

低減, X ウィンドウシステムのネットワークトラフィック, 76-77
.*ディレクトリ, 「隠しディレクトリ」を参照
ディレクトリエントリファイル, 36
テーマ
CPU の使用, 70
アイコンオプション
インストール, 44
概要, 41
新しいアイコンのインストール, 45-46
ウィンドウフレームオプション
CPU の使用, 70-71
ネットワークトラフィック, 76
ウィンドウフレームのオプション
インストール, 43-44
概要, 41
変更, 71
概要, 41
コントロールオプション
CPU の使用, 70
インストール, 43
概要, 41
カスタムの作成, 46
変更, 70
ネットワークトラフィック, 76

デスクトップ, フォント設定、GConf による設定, 28
デスクトップエントリファイル, 34-35
デスクトップの背景, オフにする, 75

ね

ネットワークトラフィック、低減, 76-77

は

背景, 設定、GConf による設定, 28
パターンマスク, 53
パネル
アニメーションをオフにする, 72
設定、個別、GConf による設定, 23-27
パフォーマンス, 改善, 概要, 69
パフォーマンスの改善, CPU 使用率を減らす, 69-76

ふ

ファイルタイプ判別パターン, 52-53
ファイル抽象レイヤー, メニュー, 32
ファイルマネージャ
サイド区画をオフにする, 74-75
ツールバー、オフにする, 74
デスクトップの背景、オフにする, 75
パフォーマンス設定の変更, 72-74
メニューの追加, 37
ロケーションバー、オフにする, 74
フォント
GConf による設定, 27-28
概要, 47
個別ユーザーに追加, 48-49
すべてのユーザーに追加, 48
置換, 47-48
フォントの設定, 「フォント」を参照

へ

減らす, CPU 使用率, 69-76
ヘルプ
OMF ファイル, 67-68

ヘルプ (続き)

ScrollKeeper カタログ作成システム, 68

概要, 67

め

メニュー

vfolder, 32-34

vfolder 情報ファイル, 33

概要, 31

項目の追加, 38

設定の編集, 38

ファイル抽象レイヤー, 32

ファイルマネージャによる追加, 37

編集, 36-39

メニュー項目の削除, 39

メニュー項目の編集, 38-39

メニューファイルを使用して追加, 37-38

ユーザーが変更できないメニューを設定, 39

メニューアイコン

CPU 使用率, 71

オフにする, 71

ネットワークトラフィック, 76

ろ

ロックダウン, 「機能の無効化」を参照

わ

ワークスペース, GConf による数の設定, 22-23