



Solaris 8 のシステム管理 (追補)

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

Part Number 806-6216-10
2000 年 10 月

Copyright 2000 Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, California 94303-4900 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 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、docs.sun.com、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、オムロン株式会社で開発されたソフトウェアです。(Copyright OMRON Co., Ltd. 1999 All Rights Reserved.)

「ATOK」は、株式会社ジャストシステムの登録商標です。

「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。

「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)は郵政省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。

DtComboBox 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。(© 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、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: Solaris 8 System Administration Supplement

Part No: 806-5183-10

Revision A



目次

- はじめに 5
- 1. 新規機能の概要 9
- 2. **USB** のサポート 13
 - USB の概要 13
 - よく使用される USB 関連の略語 14
 - USB バスの説明 15
 - Solaris 環境における USB について 17
 - SPARC: USB プリンタのサポート 27
- 3. リムーバブルメディアの管理機能の向上 29
 - 概要 29
 - リムーバブルメディア上の情報へのアクセス 30
 - ▼ リムーバブルメディア上の情報にアクセスするには 31
 - Jaz ドライブおよび Zip ドライブへのアクセス 32
 - リムーバブルメディアのフォーマット (rmformat) 33
 - ▼ リムーバブルメディアをフォーマットするには (rmformat) 33
 - ▼ UFS または UDFS ファイルシステム用にリムーバブルメディアをフォーマットするには 35
 - ▼ PCFS ファイルシステム用にリムーバブルメディアをフォーマットするには 36
 - ▼ リムーバブルメディア上の PCFS ファイルシステムをチェックするには 37

- ▼ リムーバブルメディア上の不良ブロックを修復するには 38
- リムーバブルメディアへの読み取りまたは書き込み保護とパスワードによる保護の適用 39
- ▼ リムーバブルメディアに書き込み保護を有効または無効にするには 39
- ▼ Iomega 媒体上で読み取りまたは書き込み保護とパスワードを有効または無効にするには 39
- 4. 一般的なシステム管理 43
 - DNLC 機能の向上 43
 - アカウントティング機能の拡張 45
- 5. **Solaris 8** のマニュアルからの変更点の概要 47
 - 『Solaris WBEM Services の管理』 47

はじめに

『Solaris 8 のシステム管理 (追補)』では、Solaris™ 8 Update リリースにおいて追加および変更された機能について説明します。ここでの説明は、すでにリリースされている Solaris 8 のマニュアルセットの内容を補足または変更するものです。Solaris のマニュアルは、Solaris 8 のこのリリースの DOCUMENTATION CD に含まれています。

注 - Solaris オペレーティング環境は、2 種類のハードウェア (プラットフォーム) 上で動作します。つまり、SPARC™ と IA (Intel アーキテクチャ) です。Solaris オペレーティング環境は、64 ビットと 32 ビットの両方のアドレス空間で動作し、IA では 32 ビットのアドレス空間でのみ動作します。このマニュアルで説明する情報は、章、節、注、箇条書き、図、表、例、またはコード例において特に明記しない限り、両方のプラットフォームおよびアドレス空間に該当します。

Sun のマニュアルの注文方法

専門書を扱うインターネットの書店 Fatbrain.com から、米国 Sun Microsystems™, Inc. (以降、Sun™ とします) 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、<http://www1.fatbrain.com/documentation/sun> の Sun Documentation Center をご覧ください。

Sun のオンラインマニュアル

<http://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。マニュアルのタイトルや特定の主題などをキーワードとして、検索を行うこともできます。

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。	.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。 system%
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:
AaBbCc123	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。

表 P-1 表記上の規則 続く

字体または記号	意味	例
「」	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`

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。

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

- このマニュアルでは、「IA」という用語は、Intel 32 ビットのプロセッサアーキテクチャを意味します。これには、Pentium、Pentium Pro、Pentium II、Pentium II Xeon、Celeron、Pentium III、Pentium III Xeon の各プロセッサ、および AMD、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。

新規機能の概要

この章では、Solaris 8 Update リリースに追加された新機能について説明します。

注 - 最新のマニュアルページを参照するには、man コマンドを使用してください。
Solaris 8 Update リリースのマニュアルページには、「Solaris 8 Reference Collection」には記載されていない新しい情報も提供されています。

表 1-1 Solaris 8 の新規機能

説明	サポート開始リリース番号
ボリューム管理の機能拡張	
リムーバルメディア管理が、DVD-ROM、Zip ドライブと Jaz ドライブ、CD-ROM、フロッピーディスクなどのリムーバルメディアを完全にサポートするようになりました。この機能の使い方については、第 3 章を参照してください。	6/00。10/00 で更新。
システム性能の機能拡張	
機能拡張されたディレクトリ名検索キャッシュ (DNLC) を使用すると、大規模なディレクトリ内のファイルにアクセスするときの性能が向上します。この機能の使い方については、43ページの「DNLC 機能の向上」を参照してください。	6/00
リソース管理の機能拡張	

表 1-1 Solaris 8 の新規機能 続く

説明	サポート開始リリース番号
<p>アカウントINGの拡張によって、一般的なアカウントINGデータのグループを表す、新しい可変長の汎用アカウントINGファイル形式が導入されました。また、カーネルがさまざまなアカウントINGファイルに記録したりソースの使用状況を構成する機能も導入されました。この機能の使い方については、45ページの「アカウントING機能の拡張」を参照してください。</p>	6/00
セキュリティ	
<p>Generic Security Service Application Programming Interface (GSS-API) は、セキュリティのフレームワークです。GSS-API を使用すると、アプリケーションは転送するデータを保護できます。GSS-API は認証、整合性、および機密性のサービスをアプリケーションに提供します。GSS-API を使用すると、各アプリケーションはセキュリティに関して全般的に「汎用」になります。つまり、どのような実際のプラットフォーム (Solaris プラットフォームなど) やセキュリティ機構 (Kerberos など) が使用されるかを知る必要がありません。これは、GSS-API を使用するアプリケーションの移植性が高くなることを意味します。</p> <p>詳細は、『GSS-API のプログラミング』を参照してください。</p>	6/00
ネットワーク	
<p>IP ネットワークマルチパスでは、ネットワークアダプタにおけるシングルポイントの障害からの復旧機能や、トラフィックのスルーットの向上をシステムに提供します。ネットワークアダプタにおいて障害が発生し、同じ IP リンクに代替アダプタが接続されている場合、システムはすべてのネットワークアクセスを障害の起きたアダプタから代替アダプタへ自動的に切り替えます。このプロセスにより、ネットワークへのアクセスの中断を防ぐことができます。また、同じ IP リンクに複数のネットワークアダプタが接続されている場合、トラフィックを複数のネットワークアダプタに分散させることにより、トラフィックのスルーットが向上します。</p> <p>詳細は、『IP ネットワークマルチパスの管理』を参照してください。</p>	10/00
<p>WBEM (Web-based enterprise management) には、複数のプラットフォームにおけるシステム、ネットワーク、デバイスの Web ベースでの管理の標準規格が含まれています。この標準化により、システム管理者は、デスクトップ、デバイス、およびネットワークの管理を行うことができます。このリリースでは、CIM Object Manager が使用するシステムプロパティの記述や、新しい Solaris_Printer およびその他の印刷定義のクラスについての記述などが追加されました。</p> <p>『Solaris WBEM Services の管理』の Solaris 8 Update リリースでの変更点については、47ページの「『Solaris WBEM Services の管理』」を参照してください。</p>	10/00

表 1-1 Solaris 8 の新規機能 続く

説明	サポート開始リリース番号
<p>モバイル IP (Internet Protocol) を使用すると、モバイルコンピュータ (ラップトップ、無線通信など) 間で情報を転送できます。モバイルコンピュータは別のネットワークに場所を変更しても、モバイルコンピュータのホームネットワークを通じてアクセスおよび通信できます。モバイル IP の Solaris の実装では IPv4 だけがサポートされます。</p> <p>詳細は、『モバイル IP の管理』を参照してください。</p>	6/00
<p>USB のサポート</p>	
<p>Solaris 印刷マネージャを使用して、USB (Universal Serial Bus) ポートを備えた SPARC システムに接続された USB プリンタを設定することができます。詳細は、27ページの「SPARC: USB プリンタのサポート」を参照してください。</p> <p>USB の概要については、13ページの「USB の概要」を参照してください。</p>	10/00
<p>アーリーアクセス</p>	
<p>このリリースでは、アーリーアクセス (EA) ディレクトリにアーリーアクセスソフトウェアが含まれています。詳細は、Solaris 8 のリリースの SOFTWARE 2 of 2 CD に含まれる各アーリーアクセスソフトウェアの README を参照してください。</p>	6/00

USB のサポート

注 - 最新のマニュアルページを参照するには、man コマンドを使用してください。Solaris 8 Update リリースのマニュアルページには、「*Solaris 8 Reference Manual Collection*」には記載されていない新しい情報も提供されています。

USB の概要

この機能は、Solaris 8 10/00 ソフトウェアリリースで更新されました。

Universal Serial Bus (USB) は PC 業界で開発された、周辺機器 (キーボード、マウス、プリンタなど) をシステムに接続するための低コストのソリューションです。

USB コネクタは 1 方向 1 種類のケーブルだけに適合するように設計されています。デバイスはハブデバイスに接続できます。ハブデバイスとは、複数のデバイス (他のハブデバイスも含む) を接続するためのものです。USB が設計された主な目的は、デバイスごとに異なる何種類ものコネクタを減らすことです。つまり、システムの背面パネルの混雑を軽減することです。他にも、USB デバイスを使用すると、次のような利点があります。

- USB デバイスはホットプラグ可能。詳細は、26ページの「USB デバイスのホットプラグ」を参照してください。
- 最大 126 台のデバイスを Solaris 環境でサポート
- 最大 12M ビット/秒のデータ転送速度をサポート
- ロースピードデバイス (1.5M ビット/秒) からフルスピードデバイス (12M ビット/秒) までをサポート

- 低コストの外部ハブを追加するだけで簡単にバスを拡張可能。また、ハブとハブを接続して、ツリートポロジを構築できます。

Sun Microsystems は次のようなシステムで USB デバイスをサポートします。

- Solaris 8 10/00 リリースが動作している Sun Blade™ 100 と Sun Blade 1000 システムは USB デバイスをサポートします。
- Sun Ray™ システムも USB デバイスをサポートします。
- Solaris 8 (Intel 版) が動作している IA システムは、キーボード、マウス、一部の大容量ストレージデバイス (Zip ドライブなど) について USB サポートを提供します。詳細は、scsa2usb (7D) のマニュアルページを参照してください。

次の表に、Solaris 環境でサポートされる USB デバイスを示します。

USB デバイス	サポートされるシステム
キーボードとマウス	Sun USB サポート付きの SPARC システム uhci (7D) コントローラに基づいた、USB バスを持つ IA システム (オンボードの USB コントローラだけがサポートされます。プラグイン式のホストコントローラ PCI カードはサポートされません。)
大容量ストレージ	SPARC と IA
プリンタ	SPARC
ハブ	SPARC と IA

よく使用される USB 関連の略語

次の表に、Solaris 環境で使用される USB の略語について説明します。USB の構成要素と略語についての詳細は、<http://www.usb.org> を参照してください。

略語	説明
USB	Universal Serial Bus
USBA	Universal Serial Bus Architecture (Solaris)

略語	説明
USBAI	USBA Client Driver Interface (Solaris)
HCD	USB ホストコントローラドライバ

USB バスの説明

USB 仕様は、ライセンス料無しで誰でも入手できます。USB 仕様は、バスとコネクタの電気的および機械的なインタフェースを定義します。

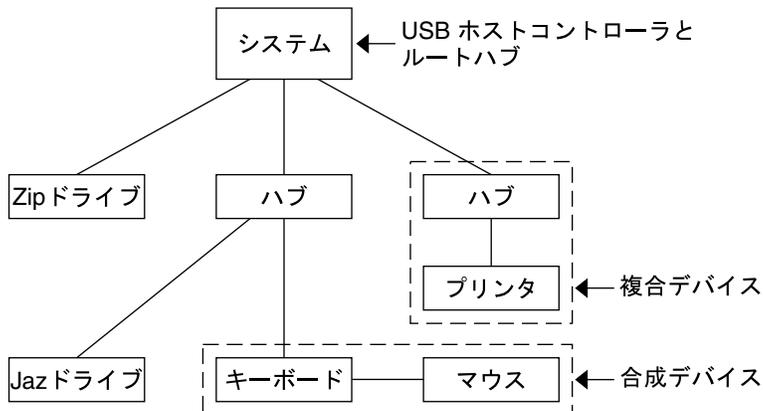


図 2-1 USB 物理デバイスの階層

USB が採用するトポロジでは、ハブが USB デバイスに接続点を提供します。ホストコントローラには、システム内のすべての USB ポートの原点となるルートハブが含まれます。ハブについての詳細は、19ページの「USB ホストコントローラとルートハブ」を参照してください。

上記の例では、1つのシステムが3つのアクティブな USB ポートを持っています。1番目の USB ポートは Zip ドライブに接続されています。この Zip ドライブにはハブが埋め込まれていないため、他のデバイスは接続できません。2番目の USB ポートはハブに接続されており、このハブには Jaz ドライブと合成デバイス (キーボードとマウス) が接続されています。このキーボードにはハブが埋め込まれており、そのハブにマウスが接続されています。

次の表に、上記の例におけるデバイスのデバイスツリーパス名を示します。

Zip ドライブ	/pci@1f,4000/usb@5/storage@1
キーボード	/pci@1f,4000/usb@5/hub@2/keyboard@1
マウス	/pci@1f,4000/usb@5/hub@2/mouse@2
Jaz ドライブ	/pci@1f,4000/usb@5/hub@2/storage@3
プリンタ	/pci@1f,4000/usb@5/hub@3/printer@1

USB デバイスとドライバ

USB デバイスはいくつかのデバイスクラスに分類されます。各デバイスクラスは対応するドライバを持っています。1つのクラス内のデバイスは同じデバイスドライバで管理されます。しかし USB 仕様では、特定のクラスに属さない、ベンダー固有のデバイスも許可しています。同じような属性とサービスを持つデバイスは同じグループにまとめます。

Human Interface Device (HID) クラスには、ユーザーが制御するデバイス (キーボード、マウス、ジョイスティックなど) が含まれます。Communication Device クラスには、電話に接続するデバイス (モデムや ISDN インタフェースなど) が含まれます。その他にも、Audio、Monitor、Printer、Storage Device などのデバイスクラスがあります。各 USB デバイスはデバイスのクラスを表す記述子を持っています。デバイスクラスは、そのメンバーが構成とデータ転送に関してどのように動作すべきかを指定します。クラス情報についての詳細は、www.usb.org サイトを参照してください。

Solaris USB Architecture (USBA)

USB デバイスは2つのレベルのデバイスツリーノードとして表現されます。1つのデバイスノードが USB デバイス全体を表し、1つまたは複数の子インタフェースノードがデバイス上にある個々の USB インタフェースを表します。特殊なケースとして、デバイスノードとインタフェースノードが1つのノードに結合される場合もあります。

ドライバのバインドは互換性のある名前属性で実現されます。詳細は、『IEEE 1275 USB binding』の3.2.2.1項と『Writing Device Drivers』を参照してください。ドライバは、デバイス全体にバインドしてすべてのインタフェースを制御することも、1つのインタフェース (キーボードやマウスなど) だけにバインドすることも可

能です。デバイス全体にバインドするドライバがベンダーにもクラスにも存在しない場合、汎用 USB マルチインタフェースドライバがデバイスレベルのノードにバインドされます。このドライバは、互換性のある名前属性 (『IEEE 1275 USB binding』の 3.2.2.1 項で定義されている) を使用して、各インタフェースにドライバをバインドしようとしています。

図 2-1 には、複合デバイス (ハブとプリンタ) の例が示されています。ハブとプリンタは両方とも同じプラスチック製のケースに入っていますが、異なる USB バスアドレスを持ちます。また、図 2-1 には、合成デバイスの例も示されています。キーボードとコントローラは同じプラスチック製のケースに入っていますが、同じ USB バスアドレスを持ちます。この例では、1 本のケーブルが USB マウスを合成デバイス (キーボードとコントローラ) に接続しています。

Solaris USB Architecture (USBA) は、USB 1.0 および 1.1 の仕様に加え、Solaris ドライバ条件に準拠しています。USBA モデルは SCSI (Sun Common SCSI Architecture) に似ています。USBA とは、汎用 USB トランスポート層という概念をクライアントドライバに提供する薄い層のことです。

SCSI と USBA の違いは、SCSI がバスを検査するときに `.conf` ファイルを使用するのに対して、USB ハブドライバは自己検査 `nexus` ドライバであることです。

Solaris 環境における USB について

次の節では、Solaris 環境における USB について知っておくべきことを説明します。

USB キーボードとマウス

Solaris 環境では複数の USB キーボードとマウスをサポートしていないため、USB キーボードとマウスは常に 1 つだけシステムに接続するようにしてください。次の項目の説明も参照してください。

- コンソールキーボードとマウスは複数のハブ経由で接続してはなりません。複数のハブを経由すると、キーボードとマウスが適時に正しく認識されず、コンソールキーボードとマウスとして構成されません。
- コンソールキーボードとマウスは、システムのリブート後は、いつでも取り外してルートハブの別のポートや外部ハブに移動できます。しかし、リブート中や `ok` プロンプトが出ている間は移動できません。キーボードをプラグインした後は、キーボードとマウスは再び完全に機能します。

- **SPARC** のみ — USB キーボードの電源キーと Sun タイプ 5 キーボードの電源キーの動作は異なります。USB キーボードでは、電源キーでシステムを中断またはシャットダウンすることができますが、システムの電源を入れることはできません。
- Sun 社製以外の USB キーボードでは、キーパッドの左側にある機能は使用できません。
- 複数のキーボードはサポートされません。
 - キーボードは認識され、使用できますが、コンソールキーボードとしては認識されません。
 - ブート時に最初に認識されたキーボードがコンソールキーボードとなります。このため、ブート時に複数のキーボードがプラグインされていると混乱の原因となります。
 - コンソールキーボードを取り外した場合、次に利用可能な USB キーボードがコンソールキーボードにはなりません。次にホットプラグされるキーボードがコンソールキーボードになります。
- 複数のマウスはサポートされません。
 - マウスは認識され、使用できますが、コンソールマウスとしては認識されません。
 - ブート時に最初に認識されたマウスがコンソールマウスとなります。このため、ブート時に複数のマウスがプラグインされていると混乱の原因となります。
 - コンソールマウスを取り外した場合、次に利用可能な USB マウスがコンソールマウスにはなりません。次にホットプラグされるマウスがコンソールマウスになります。
- Sun 社製以外の合成キーボードを PS/2 マウスと使用する場合、このキーボードがブート時に最初に認識されると、PS/2 マウスがプラグインされていなくても、このキーボードとマウスがコンソールキーボードとマウスになります。つまり、別の USB マウスがシステムにプラグインされていても、コンソールマウスとして構成されないので機能しません。
- 2 ボタンと 3 ボタンのマウスだけがサポートされます。ホイール付きマウスは 1 ボタンのマウスのように動作します。3 ボタンよりも多いマウスは 3 ボタンのマウスのように動作します。

USB ホストコントローラとルートハブ

USB ハブは次のことを行います。

- ポートにおけるデバイスの取り付けと取り外しのモニタ
- ポートにおける個々のデバイスの電源管理
- ポートへの電源の制御

USB ホストコントローラはルートハブという埋め込みハブを持っています。背面パネルに見えるポートはルートハブのポートです。USB ホストコントローラは次のことを行います。

- USB バスの指示。個々のデバイスはバスの調整はできません。
- デバイスによって決定されるポーリング間隔による、デバイスのポーリング。デバイスはポーリング間隔 (時間) の考慮に十分なバッファを持っていると想定されます。
- USB ホストコントローラとそれに接続されているデバイス間でのデータの送信。ピアツーピア通信はサポートされません。

USB ハブデバイス

- SPARC と IA のどちらのシステムにおいても、ハブを 4 段を超えて多段接続してはなりません。SPARC システムでは、Open Boot PROM (OBP) は 4 段を超えるデバイスを正確に認識できません。
- バス電源供給方式のハブは多段接続してはなりません。つまり、バス電源供給方式のハブを別のバス電源供給方式のハブに接続してはなりません。バス電源供給方式のハブは独自の電源を持っていません。USB フロッピーディスクデバイスはすべての電源をバスから取り入れるため、バス電源供給方式のハブ上では機能しません。

USB ストレージデバイス

Solaris 8 10/00 リリースでは、USB の Zip、Jaz、Clik、SmartMedia、CompactFlash、および ORB のリムーバブル大容量ストレージデバイスをサポートします。Solaris 環境でサポートされるデバイスの完全なリストについては、scsa2usb(7D) のマニュアルページを参照してください。

これらのデバイスは、ボリューム管理を使用しても使用しなくても管理することができます。ボリューム管理を実行している状態でのデバイス管理についての情報は、`vold(1M)` のマニュアルページを参照してください。

vold を実行している状態で **USB 大容量ストレージデバイス** を管理する

Solaris 共通デスクトップ環境 (CDE) が実行されている場合は、USB リムーバブル大容量ストレージデバイスは、CDE ファイルマネージャのコンポーネントであるリムーバブルメディア・マネージャによって管理されます。CDE ファイルマネージャについての詳細は、`dtfile(1)` のマニュアルページを参照してください。

注・この節に記載されているマニュアルページを表示するには、`MANPATH` 変数に `/usr/dt/man` を含める必要があります。これらのコマンドを使用するには、パスに `/usr/dt/bin` を含め、さらに CDE を実行している必要があります。また、これらのコマンドをリモートで使用する場合は、`DISPLAY` 変数を設定する必要があります。

次の表に、リムーバブルメディア・マネージャが CDE 環境からストレージデバイスを管理する際に使用するコマンドを示します。

リムーバブルメディア・マネージャが使用するコマンド	実行内容
<code>sdtmedia_format(1)</code>	USB デバイスのフォーマットおよびラベル付けを行う
<code>sdtmedia_prop(1)</code>	デバイスのプロパティを表示する
<code>sdtmedia_prot(1)</code>	デバイスのアクセス権を変更する
<code>sdtmedia_slice(1)</code>	デバイス上のスライスを作成または変更する

USB デバイスのフォーマットが終了すると、通常は `/rmdisk/label` ディレクトリの下にマウントされます。リムーバブルストレージデバイスの構成についての詳細は、`rmmount.conf(4)` または `vold.conf(4)` のマニュアルページを参照してください。

以下の手順は、ボリューム管理を実行している状態で USB 大容量ストレージデバイスを管理する方法を示しています。デバイスノードは /vol/dev ディレクトリの下に作成されます。詳細は、scsa2usb(7D) のマニュアルページを参照してください。以下の手順では、ホットプラグ可能な USB 大容量ストレージデバイスを追加および取り外す方法についても記載しています。デバイスのホットプラグとは、オペレーティングシステムをシャットダウンすることなくあるいはシステムの電源を切ることなく、デバイスを追加したり取り外したりすることを指します。

vold を実行している状態で **USB 大容量ストレージデバイス** をマウントまたはマウント解除するには

1. すべてのリムーバブル大容量ストレージデバイス (**USB 大容量ストレージデバイス** を含む) について、デバイスの別名を表示します。

```
$ eject -n
.
.
.
rmdisk0 -> /vol/dev/rdsk/c4t0d0/clik40      (Generic USB storage)
cdrom0 -> /vol/dev/rdsk/c0t6d0/audio_cd    (Generic CD device)
zip1 -> /vol/dev/rdsk/c2t0d0/fat32        (USB Zip device)
zip0 -> /vol/dev/rdsk/c1t0d0/zip100       (USB Zip device)
jaz0 -> /vol/dev/rdsk/c3t0d0/jaz1gb       (USB Jaz device)
```

2. 上記のようにリストされたデバイス別名を使用して、**USB 大容量ストレージデバイス** をマウントします。

```
$ volrmmount -i device-alias
```

次の例では、USB Jaz ドライブを /rmdisk/jaz0 の下にマウントします。

```
$ volrmmount -i jaz0
```

3. **USB** ストレージデバイスをマウント解除します。

```
$ volrmmount -e device-alias
```

次の例では、USB Zip ドライブを /rmdisk/zip0 からマウント解除します。

```
$ volrmmount -e zip0
```

4. **USB** デバイスを、汎用の **USB** ドライブから取り出します。

```
$ eject device-alias
```

次に例を示します。

```
$ eject rmdisk0
```

注 - eject コマンドは、デバイスがまだマウント解除されていない場合は、そのデバイスのマウント解除も行います。また、eject コマンドは、そのデバイスにアクセスしているアクティブなアプリケーションがある場合は、それらを終了させます。

vold の実行中にホットプラグ可能な **USB** 大容量ストレージデバイスを取り外すには

以下の手順では、vold の実行中にホットプラグ可能な USB デバイスを取り外す例として、Zip ドライブを使用しています。

1. デバイスをマウント解除します。

```
$ volrmmount -e zip0
```

2. そのデバイスを使用しているアクティブなアプリケーションがある場合は、それらを終了させます。

3. デバイスを取り出します。

```
$ eject zip0
```

4. スーパーユーザーになり、vold を停止します。

```
# /etc/init.d/volmgt stop
```

5. **USB** 大容量ストレージデバイスを取り外します。

6. `vold` を開始します。

```
# /etc/init.d/volmgt start
```

`vold` の実行中にホットプラグ可能な **USB** 大容量ストレージデバイスを追加するには

以下の手順は、`vold` の実行中にホットプラグ可能な USB デバイスを追加する方法を示します。

1. `vold` を停止します。

```
# /etc/init.d/volmgt stop
```

2. **USB** 大容量ストレージデバイスを追加します。

3. `vold` を開始します。

```
# /etc/init.d/volmgt start
```

4. デバイスが追加されたことを確認します。

```
$ ls device-alias
```

`vold` を使用せずに **USB** 大容量ストレージデバイスを管理する

USB 大容量ストレージデバイスは、ボリュームマネージャ (`vold`) を使用せずに管理することもできます。ボリュームマネージャを使用しないようにする方法は、次の 2 とおあります。

■ 次のコマンド実行して、`vold` を停止させます。

```
# /etc/init.d/volmgt stop
```

- `vold` は実行した状態にしておきますが、USB 大容量ストレージデバイスを `vold` に登録しないようにします。 `/etc/vold.conf` ファイル内の次の行でテキストの初めに `#` マークを挿入し、コメント扱いにします。

```
# use rmdisk drive /dev/rdisk/c*s2 dev_rmdisk.so rmdisk%d
```

上記のようにコメント扱いにした後、`vold(1M)` を再度開始します。

```
# pkill -HUP vold_pid
```

詳細は、`vold.conf(4)` のマニュアルページを参照してください。

`vold` を使用せずに **USB** 大容量ストレージデバイスを管理するには

以下の手順は、`vold(1M)` を使用せずに USB 大容量ストレージデバイスを管理する方法を示しています。デバイスノードは、文字型デバイスについては `/dev/rdsk` ディレクトリ、ブロック型デバイスについては `/dev/dsk` ディレクトリの下に作成されます。詳細は、`scsa2usb(7D)` のマニュアルページを参照してください。

`vold` を使用せずに **USB** 大容量ストレージデバイスをマウントまたはマウント解除するには

1. スーパーユーザーになります。
2. **USB** 大容量ストレージデバイスをマウントします。

```
# mount -F fs-type /dev/rdsk/cntndnsn /mount-point
```

3. **USB** 大容量ストレージデバイスをマウント解除します。

```
# umount /mount-point
```

4. デバイスを取り出します。

```
# eject /dev/[r]dsk/cntndnsn
```

vold を使用せずにホットプラグ可能な **USB** 大容量ストレージ デバイスを取り外すには

以下の手順は、vold を使用せずにホットプラグ可能な USB デバイスを取り外す方法を示しています。

1. スーパーユーザーになります。
2. ホットプラグ可能な **USB** デバイスを取り外します。
 - a. デバイスをマウント解除します。

```
# umount /mount-point
```

- b. そのデバイスを使用しているアクティブなアプリケーションがある場合は、それらを停止します。
- c. デバイスを取り外します。

vold を使用せずにホットプラグ可能な **USB** 大容量ストレージ デバイスを追加するには

以下の手順は、vold を使用せずにホットプラグ可能な USB デバイスを追加する方法を示しています。

1. ホットプラグ可能な **USB** デバイスを **USB** ポートに追加します。
2. **USB** デバイスが追加されたことを確認します。

```
$ ls /dev/rdisk/cntndnsm
```

SPARC のみ: **USB** 電源管理

システムが電源管理を有効にしている場合、USB のフレームワークはすべてのデバイスを電源管理しようと最大限に努力します。USB デバイスの電源管理には、デバイスが接続されているポートのハブドライバによる中断も含まれます。リモートウェイクアップ (呼び起こし) をサポートするかしないかは、デバイスによって異なります。デバイスがリモートウェイクアップをサポートしている場合は、イベントの発生時 (たとえば、マウスが移動したときなど) に、接続されているハブ

をウェイクアップします。アプリケーションが入出力を送信した場合も、ホストシステムはデバイスをウェイクアップできます。

リモートウェイクアップ機能がサポートされている場合、すべての HID (キーボードやマウスなど)、ハブ、およびストレージデバイスは、デフォルトで電源管理されます。USB プリンタが電源管理されるのは、2つの印刷ジョブ間だけです。

電源管理で電源消費を抑えるときは、まず、USB リーフデバイスの電源が切断され、しばらくしてから、親ハブの電源が切断されます。当該ハブのポートに接続されているすべてのデバイスの電源が切断されると、しばらくしてから、ハブの電源が切断されます。最も効率的な電源管理は、あまり多くのハブを多段接続しないことです。

USB デバイスのホットプラグ

USB デバイスは、プラグインするとすぐにシステムのデバイス階層に表示されます (`prtconf (1M)` コマンドで確認)。また、デバイスが使用中でない限り、USB デバイスを取り外すとシステムのデバイス階層から消えます。

使用中の USB デバイスを取り外した場合、ホットプラグの動作は少しだけ異なります。使用中の USB デバイスを取り外した場合、デバイスノードは残り、このデバイスを制御しているドライバはデバイス上のすべての動作を停止します。それ以降、このデバイスに発行される新しい入出力動作はエラーで戻されます。

このような場合、システムは元のデバイスを接続するようにユーザーにプロンプトを表示します。間違って使用中の USB デバイスを取り外してしまった場合は、次のようにして回復します。

1. 元のデバイスを同じポートに接続します。
2. そのデバイスを使用しているアプリケーションを停止します。
3. デバイスを取り外します。

元のデバイスが再びプラグインされるまで、USB ポートは使用できません。デバイスが使用できない場合は、USB ポートは次にリブートするまで使用できません。

注 - アクティブな、つまり開いているデバイスを削除すると、データの整合性が損なわれる可能性があります。デバイスを取り外す前には、必ず、デバイスを閉じるようにしてください。ただし、コンソールキーボードとマウスは例外で、アクティブなときでも移動することができます。

USB ケーブル

市販されている USB ケーブルエクステンダは絶対に使用しないでください。デバイスを接続するときは、必ず、ハブと十分な長さのあるケーブルを使用してください。USB デバイスを接続するときは、必ず、フルレイト (12M ビット/秒) の 20/28 AWG ケーブルを使用してください。

SPARC: USB プリンタのサポート

Solaris 8 10/00 リリースが動作している USB ポート付きの SPARC システムに接続されている USB プリンタでは、Solaris の印刷マネージャを使用して USB プリンタを設定できるようになりました。

USB プリンタ用の新しい論理デバイス名は次のとおりです。

```
/dev/printers/[1-9]*
```

したがって、USB プリンタをプリンタサーバーに追加するときは、「新しいローカルプリンタを設定」画面の「プリンタポート」で、上記デバイスの 1 つを USB プリンタ用に選択します。Solaris 印刷マネージャを使用してプリンタを設定する方法についての詳細は、『Solaris のシステム管理 (第 2 巻)』を参照してください。

新しい Solaris USB プリンタドライバは USB プリンタクラス準拠のプリンタをすべてサポートしますが、推奨される PostScript™ プリンタについては、usbprn (7D) のマニュアルページを参照してください。

usbprn ドライバは PostScript 以外のプリンタ (つまり、GhostScript などの Sun 社製以外の PostScript 変換パッケージを使用するプリンタ) にも準拠しています。変換パッケージは Solaris 8 Software Companion CD から入手できます。Solaris 8 Software Companion CD の入手方法については、<http://www.sun.com/software/solaris/binaries/package.html> を参照してください。

USB プリンタのホットプラグに関する情報と注意については、usbprn (7D) のマニュアルページの「NOTES」と「DIAGNOSTICS」の節を参照してください。

リムーバブルメディアの管理機能の向上

注 - 最新のマニュアルページを参照するには、man コマンドを使用してください。Solaris 8 Update リリースのマニュアルページには、「Solaris 8 Reference Manual Collection」には記載されていない新しい情報も提供されています。

概要

この機能は Solaris 8 6/00 ソフトウェアリリースで更新され、このマニュアルでは Solaris 8 6/00 リリースおよび Solaris 8 10/00 リリースで更新されました。

Solaris 8 6/00 ソフトウェアリリースではボリューム管理機能が向上し、リムーバブルメディア (着脱式媒体) を完全にサポートするようになりました。この拡張により、DVD-ROM、Iomega と USB (Universal Serial Bus) の Zip ドライブと Jaz ドライブ、CD-ROM、およびフロッピーディスクは挿入時にマウントされ、読み取ることができるようになりました。

共通デスクトップ環境 (CDE) のボリューム管理と Solaris のコマンド行機能が Solaris 8 6/00 リリースで更新されました。このマニュアルで説明する情報は、『Solaris のシステム管理 (第 1 巻)』の「CD とフロッピーディスクの使用法 (概要)」におけるリムーバブルメディアの管理についての情報に追加されるものです。ファイルマネージャでこの機能を管理する方法については、『Solaris 8 ユーザーズガイド (追補)』の「リムーバブルメディア・マネージャの使用」を参照してください。

ボリューム管理の機能の向上によって、次のことが可能になりました。

- リムーバブルメディアのフォーマット、ラベル付け、および読み取りまたは書き込みソフトウェア保護の設定。新しい `rmformat` コマンドを使用します。このコマンドは以前にリムーバブルメディアのフォーマットに使用していた `fdformat` コマンドに代わって使用するものです。
- リムーバブルメディアへの PCFS ファイルシステムの作成と検証。 `mkfs_pcfs` コマンドと `fsck_pcfs` コマンドを使用します。
- SPARC ベースのシステム上のリムーバブルメディアへの `fdisk` パーティションと PCFS ファイルシステムの作成。 IA ベースのシステムへのデータ転送を容易にします。

リムーバブルメディアを使用する場合、次のガイドラインがあります。

- DVD 媒体間でデータを転送するときは、UDFS と PCFS を使用します。
- 書き換え可能媒体 (UFS ファイルシステムを持つ PCMCIA メモリーカードやフロッピーディスクなど) 間でファイルを転送するときは、`tar` または `cpio` コマンドを使用します。 SPARC システム上に作成された UFS ファイルシステムは、 IA システム上に作成された PCMCIA またはフロッピーディスク上の UFS ファイルシステムとは異なります。
- Jaz ドライブや Zip ドライブ、またはフロッピーディスク上の重要なファイルを保護するには、書き込み保護を設定します。 Iomega 媒体にはパスワードを適用します。

リムーバブルメディア上の情報へのアクセス

リムーバブルメディア上の情報にアクセスするには、ボリュームマネージャを使用する方法と使用しない方法があります。ファイルマネージャを使用してリムーバブルメディア上の情報にアクセスする方法については、『Solaris 8 ユーザーズガイド (追補)』の「リムーバブルメディア・マネージャの使用」を参照してください。

Solaris 8 6/00 リリースから、ボリュームマネージャ (`vold`) ですべてのリムーバブルメディアデバイスを動的に管理できるようになりました。これに伴い、`/dev/rdisk/cntndnsn` や `/dev/dsk/cntndnsn` などのデバイス名でのリムーバブルメディアへのアクセスは、できなくなります。

ボリュームマネージャ `vold` が実行されている場合は、CDE のリムーバブルメディア・マネージャ、または `/cdrom0`、`/floppy`、`/rmdisk`、`/jaz0`、`/zip0` な

どのボリュームマネージャのパス名を使用して、デバイスにアクセスすることができます。

また、`/vol/dev` ディレクトリ内のエントリにより、リムーバブルメディアにアクセスすることもできます。たとえば、フロッピーディスクについては、次のように実行します。

```
/vol/dev/rdiskette0/volume-name
```

CD-ROM やリムーバブルハードディスクについては、次のように実行します。

```
/vol/dev/rdsk/cntndn/volume-name
```

リムーバブルメディアのデバイスにリムーバブルメディアが挿入されている場合は、`/vol/dev` ディレクトリ内のパスへのシンボリックリンクとして `/vol/dev/aliases` ディレクトリ内にその別名が現れます。たとえば、`test` とラベル付けされたフロッピーディスクがフロッピーディスクドライブ 0 に挿入されており、`test` とラベル付けされた CD が `/dev/rdsk/c2t1d0` の CD-ROM ドライブに挿入されている場合は、次の出力が表示されます。

```
$ ls -l /vol/dev/aliases
lrwxrwxrwx 1 root root 30 May 11 12:58 cdrom0 -> /vol/dev/rdsk/c2t1d0/test
lrwxrwxrwx 1 root root 30 May 11 12:58 floppy0 -> /vol/dev/rdiskette0/test
```

デバイス名が不明な場合は、`eject -n` コマンドを使用して、すべてのリムーバブルメディアのデバイスについてデバイス名を表示することができます。たとえば、`eject -n` 出力の右側のデバイス名の一覧をもとにして、`fsck`、`mkfs`、または `newfs` コマンドで使用するデバイス名を決定することができます。

▼ リムーバブルメディア上の情報にアクセスするには

コマンド行インタフェースを使用する場合は、適切なデバイス名を使用して情報にアクセスします。リムーバブルメディアにアクセスする前に `volcheck` コマンドを実行して、コマンド行からボリュームマネージャのニックネームを使用することもできます。デバイス名については、`rmformat(1)` のマニュアルページを参照してください。

例 — リムーバブルメディア上の情報にアクセスする

フロッピーディスク上の情報にアクセスするには、次のようにします。

```
$ volcheck
$ ls /floppy
myfile
```

Jaz ドライブ上の情報にアクセスするには、次のようにします。

```
$ volcheck
$ ls /rmdisk
jaz0/ jaz1/
```

CD-ROM 上の情報にアクセスするには、次のようにします。

```
$ volcheck
$ ls /cdrom
solaris_8_sparc/
```

Jaz ドライブおよび Zip ドライブへのアクセス

Solaris 8 6/00 リリースにアップグレードするか Solaris 8 6/00 リリースをインストールするかによって、Solaris 8 6/00 で更新された Jaz ドライブおよび Zip ドライブへのアクセス方法が異なります。

- 以前の Solaris リリースから Solaris 8 6/00 リリースへアップグレードする場合は、以前の Solaris リリースと同じ方法で Jaz ドライブおよび Zip ドライブにアクセスすることができます。
- 新規に Solaris 8 6/00 リリースをインストールする場合は、以前の Solaris リリースと同じ方法で Jaz ドライブおよび Zip ドライブにアクセスすることはできません。

以前の Solaris リリースと同じ方法で Jaz ドライブおよび Zip ドライブにアクセスしたい場合は、次の手順に従ってください。

1. スーパーユーザーになります。
2. /etc/vold.conf ファイル内の次の行でテキストの初めに # マークを挿入し、コメント扱いにします。

```
# use rmdisk drive /dev/rdisk/c*s2 dev_rmdisk.so rmdisk%d
```

3. システムをリブートします。

```
# init 6
```

リムーバブルメディアのフォーマット (rmformat)

リムーバブルメディアをフォーマットするには、`rmformat` コマンドを使用します。次の種類のフロッピーディスクもこのコマンドでフォーマットできます。

- 倍密度 — 720K バイト (3.5 インチ)
- 高密度 — 1.44M バイト (3.5 インチ)

`rmformat` コマンドは非スーパーユーザーのユーティリティであり、書き込み可能なリムーバブルメディアをフォーマットおよび保護できます。`rmformat` コマンドには3つのフォーマットオプションがあります。

- `quick` — トラックを検証せずに、あるいは、検証するトラックを制限して、リムーバブルメディアをフォーマットします。
- `long` — リムーバブルメディアを完全にフォーマットします。デバイスによっては、ドライブ自身による媒体全体の検証も含まれる場合があります。
- `force` — ユーザーへの確認なしに、リムーバブルメディアを完全にフォーマットします。パスワードによる保護機能を備えた媒体では、パスワードはフォーマットする前にクリアされます。この機能はパスワードを忘れてしまったときに便利です。パスワードによる保護機能を備えていない媒体では、`long` オプションのフォーマットが行われます。

▼ リムーバブルメディアをフォーマットするには (rmformat)

`rmformat` コマンドはリムーバブルメディアをフォーマットして、デフォルトで、パーティション0とパーティション2(媒体全体)の2つのパーティションを媒体上に作成します。

1. ボリュームマネージャが動作していることを確認します。つまり、デバイス名のニックネームを使用できます。

```
$ ps -ef | grep vold
root  212    1  0   Nov 03 ?           0:01 /usr/sbin/vold
```

リムーバブルメディアデバイス名を決定する方法とボリュームマネージャを(動作していない場合)起動する方法については、『Solaris のシステム管理 (第 1 巻)』を参照してください。

2. リムーバブルメディアをフォーマットします。

```
$ rmformat -F [ quick | long | force ] device-name
```

rmformat のフォーマットオプションについては、前出の節を参照してください。

rmformat の出力が不良ブロックを示している場合、下記の手順を参照して、不良ブロックを修復してください。

3. (省略可能) リムーバブルメディアに、Solaris 環境で使用する 8 文字のラベルを付けます。

```
$ rmformat -b label device-name
```

DOS ラベルを作成する方法については、mkfs_pcfs(1M) のマニュアルページを参照してください。

例 — リムーバブルメディアをフォーマットする

フロッピーディスクをフォーマットするには、次のようにします。

```
$ rmformat -F quick /dev/rdiskette
Formatting will erase all the data on disk.
Do you want to continue? (y/n) y
.....
```

Zip ドライブをフォーマットするには、次のようにします。

```
$ rmformat -F quick zip0
Formatting will erase all the data on disk.
Do you want to continue? (y/n) y
.....
```

▼ UFS または UDFS ファイルシステム用にリムーバブルメディアをフォーマットするには

1. リムーバブルメディアをフォーマットします。

```
$ rmformat -F quick device-name
```

2. (省略可能) 代替の **Solaris** パーティションテーブルを作成します。

```
$ rmformat -s slice-file device-name
```

スライスファイルの例は次のようになります。

```
slices: 0 = 0, 30MB, "wm", "home" :
         1 = 30MB, 51MB :
         2 = 0, 94MB, "wm", "backup" :
         6 = 81MB, 13MB
```

代替の Solaris パーティションテーブルを作成する方法については、『Solaris のシステム管理 (第 1 巻)』を参照してください。

3. スーパーユーザーになります。
4. 適切なファイルシステムの種類を決定して、次のうちの 1 つを選択します。
 - a. **UFS** ファイルシステムを作成します。

```
# newfs device-name
```

- b. **UDFS** ファイルシステムを作成します。

```
# mkfs -F udfs device-name
```

例 — UFS ファイルシステム用にリムーバブルメディアをフォーマットする

次の例では、フロッピーディスクをフォーマットして、UFS ファイルシステムを作成します。

```
$ rmformat -F quick /dev/rdiskette
Formatting will erase all the data on disk.
Do you want to continue? (y/n)y
$ su
# newfs /dev/rdiskette
newfs: construct a new file system /dev/rdiskette: (y/n)? y
/dev/rdiskette: 2880 sectors in 80 cylinders of 2 tracks, 18 sectors
      1.4MB in 5 cyl groups (16 c/g, 0.28MB/g, 128 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
      32, 640, 1184, 1792, 2336,
#
```

▼ PCFS ファイルシステム用にリムーバブルメディアをフォーマットするには

1. リムーバブルメディアをフォーマットします。

```
$ rmformat -F quick device-name
```

2. スーパーユーザーになります。

3. (省略可能) 代替の **Solaris** fdisk パーティションテーブルを作成します。

```
# fdisk device-name
```

fdisk パーティションを作成する方法については、『Solaris のシステム管理 (第 1 巻)』を参照してください。

4. **PCFS** ファイルシステムを作成します。

```
# mkfs -F pcfs device-name
```

例 — PCFS ファイルシステム用にリムーバブルメディアをフォーマットする

次の例では、代替 fdisk パーティションを作成して、PCFS ファイルシステムを作成します。

```
$ rmformat -F quick /dev/rdisk/c0t4d0s2:c
Formatting will erase all the data on disk.
Do you want to continue? (y/n)y
$ su
# fdisk /dev/rdisk/c0t4d0s2:c
# mkfs -F pcfs /dev/rdisk/c0t4d0s2:c
Construct a new FAT file system on /dev/rdisk/c0t4d0s2:c: (y/n)? y
#
```

次の例では、fdisk パーティションを作成せずに、PCFS ファイルシステムを作成します。

```
$ rmformat -F quick /dev/rdiskette
Formatting will erase all the data on disk.
Do you want to continue? (y/n)y
$ su
# mkfs -F pcfs -o nofdisk,size=2 /dev/rdiskette
Construct a new FAT file system on /dev/rdiskette: (y/n)? y
#
```

▼ リムーバブルメディア上の PCFS ファイルシステムをチェックするには

1. スーパーユーザーになります。
2. **PCFS** ファイルシステムをチェックします。

```
# fsck -F pcfs device-name
```

例 — リムーバブルメディア上の PCFS ファイルシステムを チェックする

```
# fsck -F pcfs /dev/rdsk/c0t4d0s2
** /dev/rdsk/c0t4d0s2
** Scanning file system meta-data
** Correcting any meta-data discrepancies
1457664 bytes.
0 bytes in bad sectors.
0 bytes in 0 directories.
0 bytes in 0 files.
1457664 bytes free.
512 bytes per allocation unit.
2847 total allocation units.
2847 available allocation units.
#
```

▼ リムーバブルメディア上の不良ブロックを修復する には

ドライブが不良ブロック管理をサポートしている場合にのみ、検証中に見つかった不良セクタを `rmformat` コマンドで検証、解析、および修復できます。ほとんどのフロッピーディスクや PCMCIA メモリーカードは不良ブロック管理をサポートしていません。

ドライブが不良ブロック管理をサポートしている場合、不良ブロックを修復するための最大の努力が行われます。それでも不良ブロックを修復できなかった場合、修復に失敗したことを示すメッセージが表示されます。

1. リムーバブルメディア上の不良ブロックを修復します。

```
$ rmformat -c block-numbers device-name
```

`block-numbers` には、前の `rmformat` セッションで獲得したブロック番号を 10 進数、8 進数、または 16 進数形式で指定します。

2. リムーバブルメディアを検証します。

```
$ rmformat -V read device-name
```

リムーバブルメディアへの読み取りまたは書き込み保護とパスワードによる保護の適用

Iomega 媒体 (Zip ドライブや Jaz ドライブなど) には、読み取りまたは書き込み保護を適用し、パスワードを設定することができます。その他の媒体では、パスワードの設定は無しで、書き込み保護を有効または無効にできます。

▼ リムーバブルメディアに書き込み保護を有効または無効にするには

1. 書き込み保護を有効または無効のどちらにするかを決定し、次のうちの 1 つを選択します。

- a. 書き込み保護を有効にします。

```
$ rmformat -w enable device-name
```

- b. 書き込み保護を無効にします。

```
$ rmformat -w disable device-name
```

2. リムーバブルメディアの書き込み保護が有効または無効であることを確認します。

```
$ rmformat -p device-name
```

▼ Iomega 媒体上で読み取りまたは書き込み保護とパスワードを有効または無効にするには

パスワードによる保護機能をサポートしている Iomega 媒体には、最大 32 文字のパスワードを適用できます。このような Iomega 媒体上で読み取りまたは書き込み保護を設定するときは、必ずパスワードを適用する必要があります。このとき、パスワードを提供するように促すプロンプトが表示されます。

パスワード機能をサポートしていないリムーバブルメディア上でパスワードを適用しようとする、警告メッセージが表示されます。

1. 読み取りまたは書き込み保護とパスワード保護を有効または無効のどちらにするかを決定します。

- a. 読み取りまたは書き込み保護を有効にします。

```
$ rmformat -W enable device-name  
Please enter password (32 chars maximum): xxx  
Please reenter password:
```

```
$ rmformat -R enable device-name  
Please enter password (32 chars maximum): xxx  
Please reenter password:
```

- b. 読み取りまたは書き込み保護を無効にし、パスワードを削除します。

```
$ rmformat -W disable device-name  
Please enter password (32 chars maximum): xxx
```

```
$ rmformat -R disable device-name  
Please enter password (32 chars maximum): xxx
```

2. リムーバブルメディアの読み取りまたは書き込み保護が有効または無効であることを確認します。

```
$ rmformat -p device-name
```

例 — 読み取りまたは書き込み保護を有効または無効にする

次の例では、Zip ドライブ上で書き込み保護を有効にし、パスワードを設定します。

```
$ rmformat -W enable /vol/dev/aliases/zip0
Please enter password (32 chars maximum): xxx
Please reenter password: xxx
```

次の例では、Zip ドライブ上で書き込み保護を無効にし、パスワードを削除します。

```
$ rmformat -W disable /vol/dev/aliases/zip0
Please enter password (32 chars maximum): xxx
```

次の例では、Zip ドライブ上で読み取り保護を有効にし、パスワードを設定します。

```
$ rmformat -R enable /vol/dev/aliases/zip0
Please enter password (32 chars maximum): xxx
Please reenter password: xxx
```

次の例では、Zip ドライブ上で読み取り保護を無効にし、パスワードを削除します。

```
$ rmformat -R disable /vol/dev/aliases/zip0
Please enter password (32 chars maximum): xxx
```


一般的なシステム管理

この章では、次のような新しいシステム管理機能について説明します。

- 43ページの「DNLC 機能の向上」
- 45ページの「アカウントिंग機能の拡張」

注 - 最新のマニュアルページを参照するには、man コマンドを使用してください。Solaris 8 Update リリースのマニュアルページには、「*Solaris 8 Reference Manual Collection*」には記載されていない新しい情報も提供されています。

DNLC 機能の向上

この機能は、Solaris 8 6/00 ソフトウェアリリースで更新されました。

ディレクトリ名検索キャッシュ (DNLC) の機能が Solaris 8 6/00 ソフトウェアリリースで拡張され、1000 以上のファイルを持つ大規模なディレクトリ内のファイルにアクセスするときの性能が向上しました。このマニュアルで説明する情報は、『Solaris のシステム管理 (第 2 巻)』の「システム性能の概要」におけるシステム性能の管理についての情報を補足するものです。

DNLC は一般的なファイルシステムサービスであり、最近参照されたディレクトリ名とそれに関連する v ノードをキャッシュに書き込みます。UFS ディレクトリエントリはディスクに直線的に格納されます。つまり、エントリを見つけるには、各エントリを名前を検索する必要があります。新しいエントリを追加する際は、ディレクトリ全体を検索して、その名前が存在していないことを確認する必要があります。

す。この性能における問題を解決するため、DNLC はディレクトリ全体をメモリー (キャッシュ) に書き込みます。

このリリースにおける DNLC のもう一つの新機能は、検索したが存在しなかったファイルオブジェクトをキャッシュに書き込むことです。これは「ネガティブキャッシング」と呼びます。ファイルが存在するかどうかを繰り返しテストするアプリケーションに便利です。

次の節では、調整可能な新しい DNLC パラメータについて説明します。このようなパラメータは最適に設定されており、通常は変更すべきではありません。

注 - MAXUINT は符号なし整数の最大値です。

`dnlc_dir_enable`

説明	大規模ディレクトリキャッシングを有効にする
データ型	符号なし整数
デフォルト値	1 (有効)
範囲	0 (無効)、1 (有効)
変更するとき	ディレクトリキャッシングに現在既存の障害はありません。しかし、問題が発生した場合は、 <code>dnlc_dir_enable</code> を 0 に設定し、キャッシングを無効にしてください。

`dnlc_dir_min_size`

説明	1 つのディレクトリに対してキャッシュできるエントリの最小数
データ型	符号なし整数
デフォルト値	40
範囲	0 から MAXUINT まで
変更するとき	小規模ディレクトリのキャッシングで性能の問題が発生した場合、 <code>dnlc_dir_min_size</code> の値を増やしてください。個々のファイルシステムがディレクトリのキャッシングについて独自の制限を持つ場合もあるので注意してください。たとえば、UFS は各エントリが 16 バイト

であると仮定し、ディレクトリの最小数を `ufs_min_dir_cache` バイト (約 1024 エントリ) に制限します。

`dnlc_dir_max_size`

説明	キャッシュされるまでのディレクトリエントリの最大数
データ型	符号なし整数
デフォルト値	MAXUINT
範囲	0 から MAXUINT まで
変更するとき	大規模ディレクトリで性能の問題が発生した場合、 <code>dnlc_dir_max_size</code> の値を減らしてください

アカウントティング機能の拡張

Solaris のアカウントティングソフトウェアが Solaris 8 6/00 リリースで更新されました。ここで説明する情報は、『Solaris のシステム管理 (第 2 巻)』の「アカウントティングの設定と管理作業」におけるシステムアカウントティングによるリソースの管理についての情報を補足するものです。

アカウントティングの拡張によって、一般的なアカウントティングデータのグループを表す、新しい可変長の汎用アカウントティングファイル形式が導入されました。また、カーネルがさまざまなアカウントティングファイルに記録したリソースの使用状況を構成する機能も導入されました。次に、拡張されたアカウントティング機能について説明します。

- タスク — リソースの使用状況を追跡するための新しいプロセス集合
- プロジェクト — リソースの使用状況を請求するための新しい管理データベース。タスクによるリソースの使用をプロジェクトに請求できます。
- `acctadm` — 拡張アカウントティング機能の様々な属性を構成するための新しいツール。たとえば、アカウントティングシステムが追跡するリソースをシステム全体に対して構成できます。

新しいデフォルトのアカウントティング構成は管理が不要で簡単です。しかし、拡張アカウントティング機能を使用する場合、`/etc/project` ファイルを削除してはな

りません。このファイルには、拡張アカウントの構成についての重要な情報が入っています

このリリースで拡張されたアカウント機能についての詳細な情報は、次の表を参照してください。

情報	参照先
拡張されたアカウント機能の停止および起動について	acctadm(1M)
プロジェクトデータベースの記述について	projects(4)
拡張されたアカウント機能のデータの処理について	libexacct(3LIB)、getacct(2)、putacct(2)、wrcacct(2)

Solaris 8 のマニュアルからの変更点の概要

何冊かの Solaris 8 のマニュアルが Solaris 8 Update リリースで更新されました。この章では、下記のマニュアルについて、Solaris 8 リリースからの変更点の概要を記載します。

注 - 最新のマニュアルページを参照するには、man コマンドを使用してください。Solaris 8 Update リリースのマニュアルページには、「Solaris 8 Reference Manual Collection」には記載されていない新しい情報も提供されています。

『Solaris WBEM Services の管理』

Solaris 8 10/00 リリースで『Solaris WBEM Services の管理』に新規に追加された情報の概要は次のとおりです。

付録 B が更新され、次の情報が含まれるようになりました。

- Solaris_CIMOM1.0.mof ファイルの記述。このファイルには、CIM Object Manager が使用するすべてのシステムプロパティの記述が含まれています。
- Solaris_Device1.0.mof ファイルが拡張され、新しい Solaris_Printer とその他の印刷定義のクラス、および Solaris_TimeZone クラスの記述が含まれるようになりました。

詳細は、『Solaris WBEM Services の管理』の「Solaris スキーマ」の章を参照してください。