

# Debian GNU/Linux 6.0 (squeeze) リリースノート (32 ビット PC 用)

## The Debian Documentation Project [<http://www.debian.org/doc/>]

**Debian GNU/Linux 6.0 (squeeze) リリースノート (32 ビット PC  
用): The Debian Documentation Project [[http://  
www.debian.org/doc/](http://www.debian.org/doc/)]**

Publication date 2010-11-12

この文書はフリーソフトウェアです。あなたは、Free Software Foundation が公表した GNU 一般公有使用許諾の第二版の条件に基づいて、本文書の再頒布および更 を行うことができます。

本プログラムはその有用性が期待されて頒布されるものですが、市場性や特定の目的への適合性に関する暗 の保証も含め、いかなる保証も行ないません。詳細については GNU 一般公有使用許諾書をご覧ください。

あなたは、このプログラムとともに、GNU 一般公有使用許諾の写しを受け取っているはずですが、そうでなければ Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA に手紙でお知らせください。

この許諾文書は <http://www.gnu.org/licenses/gpl-2.0.html> や、Debian GNU/Linux の `/usr/share/common-licenses/GPL-2` にあります。

# 目次

- 1. はじめに **xx**
  - この文書に関するバグを報告する **xx**
  - アップグレードについての報告をする **xx**
  - この文書のソース **xx**
- 2. Debian GNU/Linux 6.0 の最新情報 **xx**
  - ディストリビューションの最新情報 **xx**
    - ファームウェアは **non-free** セクションへ移動しました **xx**
    - パッケージ管理 **xx**
    - 依存関係ベースの起動 **xx**
    - キーボード設定の統一 **xx**
    - カーネルモードセッティング (kernel mode setting, KMS) **xx**
    - LDAP サポート **xx**
    - proposed-updates セクション **xx**
    - stable-updates セクション **xx**
    - backports.org/backports.debian.org **xx**
  - Debian Live **xx**
    - ニューロイメージング研究の包括的なサポート **xx**
- 3. インストールシステム **xx**
  - インストールシステムの最新情報 **xx**
    - 主要な更点 **xx**
    - 自動インストール **xx**
- 4. Debian 5.0 (lenny) からのアップグレード **xx**
  - アップグレードの準備 **xx**
    - あらゆるデータや設定情報をバックアップする **xx**
    - 事前にユーザに通知する **xx**
    - サービスのダウン期間の準備 **xx**
    - 復旧の準備 **xx**
    - アップグレード用の安全な環境の準備 **xx**
    - 衝突しているパッケージを削除する **xx**
  - システムの状態をチェックする **xx**
    - パッケージマネージャにおいて中断しているアクションの確認 **xx**
    - APT の **pin** 機能を無効にする **xx**
    - パッケージの状態をチェックする **xx**
    - proposed-updates セクション **xx**
    - 非公式なソースとバックポートパッケージ **xx**
  - APT の取得先 (ソース) の準備 **xx**
    - APT のインターネットソースの追加 **xx**
    - APT のローカルミラーソースの追加 **xx**
    - APT の CD-ROM/DVD ソースの追加 **xx**
  - パッケージのアップグレード **xx**
    - セッションの記 **xx**
    - パッケージリストの更新 **xx**
    - アップグレードするのに十分な領域があることを確認する **xx**
    - システムの最小アップグレード **xx**
    - カーネルと **udev** のアップグレード **xx**
    - システムのアップグレード **xx**
  - アップグレード中の注意点 **xx**
    - squeeze** の Linux カーネルには **cryptloop** のサポートが含まれていません **xx**
    - 予期されるパッケージの削除 **xx**
    - aptitude** あるいは **apt-get** を実行中のエラー **xx**
    - 衝突 (Conflicts) あるいは先行依存 (Pre-Depends) のループ **xx**
    - ファイルの衝突 **xx**
    - 設定の更 **xx**
    - コンソール接続へセッションの更 **xx**

- 特定のパッケージに する特別な注意 **xx**
- カーネルと関連パッケージのアップグレード **xx**
  - カーネルメタパッケージのインストール **xx**
  - デバイスの整列順序の更 **xx**
  - 起動タイミングの問題 **xx**
- システムの起動が **Waiting for root file system** とともにハングしてしまう **xx**
  - アップグレードの前に問題を防ぐには **xx**
  - アップグレードの後で問題を解決するには **xx**
- 次期リリースへの準備 **xx**
  - GRUB 2** へアップグレードする **xx**
  - 止予定 のコンポーネント **xx**
  - 時代 れ (**Obsolete**) のパッケージ **xx**
    - ダミーパッケージ **xx**
- 5. **squeeze** で知っておくべき問題点 **xx**
  - 生じうる問題 **xx**
    - IDE** から **PATA** サブシステムへのディスクドライバの移行 **xx**
    - mdadm** メタデータ形式の更 は最新の **Grub** を必要とします **xx**
    - 新しい **libdb** による **pam\_userdb.so** の非互換性 **xx**
    - /bin/sh** の切り換え (**diversion**) に関する問題の可能性 **xx**
    - リソースの競合に関連したカーネルのポリシー更 **xx**
  - LDAP** サポート **xx**
  - sieve** サービスが **IANA** 割り当てのポート番号に移動しました **xx**
  - KDE** デスクトップ **xx**
    - KDE 3** からのアップグレード **xx**
    - 新しい **KDE** メタパッケージ **xx**
  - GNOME** デスクトップに関する更 とサポート **xx**
    - GDM 2.20** と **2.30** **xx**
    - デバイスその他の管理者限 **xx**
    - network-manager** と **ifupdown** 間での動作の関係 **xx**
  - グラフィック関連ソフトウェアの更 **xx**
    - 時代 れ (**Obsolete**) となった **Xorg** ドライバ **xx**
    - カーネルモードセッティング (**Kernel mode setting**) **xx**
    - 入力デバイスのホットプラグ **xx**
    - X server** の 「停止方法」 **xx**
- 6. **Debian GNU/Linux** に関するさらなる情報 **xx**
  - もっと読みたい **xx**
  - 助けを求めるには **xx**
    - メーリングリスト **xx**
    - インターネットリレーチャット (**IRC**) **xx**
  - バグを報告する **xx**
  - Debian** に貢献する **xx**
- A. アップグレードの前に **lenny** システムを調整する **xx**
  - lenny** システムのアップグレード **xx**
  - ソースリストのチェック **xx**
  - 古いロケールから **UTF-8** へアップグレードする **xx**
- B. リリースノートの貢献者たち **xx**
  - 目次 **xx**
  - 用語集 **xx**

# 第1章 はじめに

この文書は Debian GNU/Linux ディストリビューションのユーザに、バージョン 6.0 (コードネーム `squeeze`) での大きな更点を知らせるものです。

このリリースノートでは、リリース 5.0 (コードネーム `lenny`) から今回のリリースへの安全なアップグレード方法や、その際ユーザが遭遇する可能性がある既知の問題点について、情報をユーザに提供しています。

この文書の最新版は、<http://www.debian.org/releases/squeeze/releasenotes> から取得できます。疑わしい場合は、最初のページにある文書の日付をチェックし、最新版を読んでいるかを確認してください。

## 注意

既知の問題点をすべてリストアップすることは不可能なので、問題点の予想される広がり具合と影響の大ききの両方に基づいて取捨選 していることに注意してください。

Debian の 1 つ前のリリースからのアップグレード (この場合、5.0 からのアップグレード) のみがサポート・記述されていることに注意してください。さらに古いリリースからのアップグレードが必要な場合は、過去のリリースノートを読み、まず 5.0 へとアップグレードすることをお勧めします。

## この文書に関するバグを報告する

私たちは、この文書で明 されているすべての異なるアップグレード手順を試し、また、ユーザが直面する可能性のある、すべての問題を予想しました。

そうはいつても、この文書にバグ (不正確な情報や け落ちている情報) を見つけたと思う場合は、`release-notes` パッケージに するバグ報告として、バグ追跡システム [<http://bugs.debian.org/>]に提出してください。あなたが発見した問題が既に報告されている場合に備え、まずは既存のバグ報告 [<http://bugs.debian.org/release-notes>]を確認してみると良いでしょう。もしこの文書にさらに内容を付加できるのであれば、既存のバグ報告への情報の追加はご自由になさってください。

私たちは、この文書のソースへのパッチを含めた報告を迎・推 します。このドキュメントのソースの取得方法の記述については この文書のソース項 で、より詳細な情報を見つけてことができるでしょう。

## アップグレードについての報告をする

`lenny` から `squeeze` へのアップグレードに関連するユーザからの情報はどんなものでも迎 します。情報を共有するのを厭わない場合は、`upgrade-reports` パッケージに するバグ報告として、アップグレードの結果を含めてバグ追跡システム [<http://bugs.debian.org/>]に提出してください。報告に添付ファイルを含める場合は、(`gzip` を使用して) 縮 するようお願いします。

アップグレードについての報告を提出する際には、以下の情報を含めてください。

アップグレード前後のパッケージデータベースの状態。`/var/lib/dpkg/status` にある `dpkg` の状態データベースと、`/var/lib/apt/extended_states` にある `apt` のパッケージ状態情報です。あらゆるデータや設定情報をバックアップする項で明 するように、アップグレードを実行する前にバックアップをとっておくべきですが、`/var/lib/dpkg/status` のバックアップは `/var/backups` にもあります。`script` を使用して作成したセッションのログ。セッションの記項 で明 します。`/var/log/apt/term.log` にある `apt` のログか、`/var/log/aptitude` にある `aptitude` のログ。

## 注意

バグ報告に情報を含める前に、慎重に扱うべき情報や機密情報がログに含まれていないかある程度時間をかけてし、ログから削除してください。なぜなら、バグ報告に含まれる情報は公開データベースで公表されるからです。

## この文書のソース

この文書のソースは DocBook XML 形式です。HTML 版は、`docbook-xsl` と `xsltproc` を使用して生成しています。PDF 版は、`dblatex` や `xmlroff` を使用して生成しています。リリースノートのソースは *Debian* ドキュメンテーションプロジェクト (*Debian Documentation Project*) の SVN リポジトリにあります。ウェブからウェブインターフェース [<http://svn.debian.org/viewsvn/ddp/manuals/trunk/release-notes/>]を使って個々のファイルにアクセスでき、更を参照できます。SVN へのアクセス方法に関してさらに詳しく知りたい場合は、*Debian* ドキュメンテーションプロジェクトの SVN 情報ページ [<http://www.debian.org/doc/cvs>]を参照してください。

# 第2章 Debian GNU/Linux 6.0 の最新情報

このトピックに関するより詳しい情報が Wiki [<http://wiki.debian.org/NewInSqueeze>] に載っています。

このリリースでは、HP PA-RISC ('hppa') [<http://lists.debian.org/debian-develop-announce/2010/09/msg00008.html>]、Alpha ('alpha')、ARM ('arm') アーキテクチャへの公式サポートを止しました。

Debian GNU/Linux squeeze で公式にサポートされているアーキテクチャは以下のとおりです。

- 32 ビット PC ('i386')
- SPARC ('sparc')
- PowerPC ('powerpc')
- MIPS ('mips' (ビッグエンディアン) および 'mipsel' (リトルエンディアン))
- Intel Itanium ('ia64')
- S/390 ('s390')
- 64 ビット PC ('amd64')
- ARM EABI ('armel')

公式にサポートされているアーキテクチャに加えて、Debian GNU/Linux squeeze ではテクノロジープレビューとして GNU/kFreeBSD 移植版 ('kfreebsd-amd64' および 'kfreebsd-i386') を導入しました。これらの移植版は、Linux カーネルではなく FreeBSD カーネルと GNU ユーザランドをベースにした、Debian のリリースに含められた初めてのものになります。しかし、このバージョンのユーザは、この移植版の品質は、非常に高い品質の Linux 移植版にまだ追いつこうとしている最中であり、まだ先進的なデスクトップの機能のいくつかはサポートされていないことに注意が必要です。ですが、一般的なサーバソフトウェアのサポートは強力であり、BSD 界限で知られたユニークな機能によって、Linux ベースの Debian バージョンの機能をより広げます。これは Linux ディストリビューションが Linux ではないカーネルの利用も行えるよう張された、初めての例です。

移植状況の詳細や、お使いの移植版に特有の情報については、Debian の移植版に関するウェブページ [<http://www.debian.org/ports/>] で読むことができます。

## ディストリビューションの最新情報

Debian のこの新しいリリースには、一つ前のリリースである lenny に含まれていたよりさらに多くのソフトウェアが含まれています。このディストリビューションには、10352 以上の新しいパッケージが含まれており、全体のパッケージ数は 29050 以上になりました。ディストリビューション中のほとんどのソフトウェア、すなわち約 15436 ものソフトウェアパッケージ (これは lenny のパッケージ全体の 67% にあたります) が更新されました。また、かなりの数のパッケージ (lenny のパッケージの 18% にあたる 4238 以上) が、々な理由でディストリビューションから取り除かれました。これらのパッケージは更新されることはなく、パッケージ管理用のフロントエンドでは 'obsolete' というマークが付けられます。

このリリースで、Debian GNU/Linux では X.Org 7.3 を X.Org 7.5 に更新しました。

今回も Debian GNU/Linux にはいくつかのデスクトップアプリケーションやデスクトップ環境が含まれています。特に、GNOME 2.30、KDE 4.4.5、Xfce 4.6.2、および LXDE 0.5.0 です。生産性アプリケーションも更新されました。オフィススイートの OpenOffice.org 3.2.1 と KOffice 2.2.1、それに GNUcash 2.2.9、GNUmeric 1.10.8、Abiword 2.8.2 などです。

他のデスクトップアプリケーションも更新されました。Evolution 2.30.3 や Pidgin 2.7.3 などです。Mozilla スイートも更新されました: iceweasel (バージョン 3.5.13) は Firefox ウェブブラウザの非ブランド版で、icedove (バージョン 3.0.7) は Thunderbird メールクライアントの非ブランド版です。

またこのリリースには、特に げるなら、以下のソフトウェアの更新も含まれています:

パッケージ	5.0 ( <b>lenny</b> ) でのバージョン	6.0 ( <b>squeeze</b> ) でのバージョン
Apache	2.2.9	2.2.16
BIND DNS サーバ	9.6.0	9.7.1
Cherokee ウェブサーバ	0.7.2	1.0.8
Courier MTA	0.60.0	0.63.0
Dia	0.96.1	0.97.1
Ekiga VoIP クライアント	2.0.12	3.2.7
Exim デフォルトの電子メールサーバ	4.69	4.72
GNU Compiler Collection (デフォルトのコンパイラ)	4.3.2	4.4.5
GIMP	2.4.7	2.6.10
GNU C ライブラリ	2.7	2.11.2
lighttpd	1.4.19	1.4.28
maradns	1.3.07.09	1.4.03
MySQL	5.0.51a	5.1.49
OpenLDAP	2.4.11	2.4.23
OpenSSH	5.1p1	5.5p1
PHP	5.2.6	5.3.2
Postfix MTA	2.5.5	2.7.1
PostgreSQL	8.3.5	8.4.5
Python	2.5.2	2.6.6
Samba	3.2.5	3.5.5
Tomcat	5.5.26	6.0.28

公式の Debian GNU/Linux ディストリビューションは、現在は 4 5 枚のバイナリ DVD または 28 32 枚のバイナリ CD (アーキテクチャによる) および 4 枚のソース DVD または 28 枚のソース CD の形で提供されています。加えて、amd64 および i386 アーキテクチャのリリースのサブセットおよびそのソースコードを収めた *multi-arch* DVD もあります。Debian GNU/Linux は Blu-ray イメージの形でリリースしています。これもまた amd64 および i386 アーキテクチャとそのソースコードが収められています。

Debian は、Linux Standards Base (LSB) のバージョン 3.2 をサポートし続けています。

ファームウェアは **non-free** セクションへ移動しました

Linux カーネルに含まれる一部のドライバには、**non-free** なファームウェア **blob** が含まれていました。**squeeze** から、このファームウェアはアーカイブの **non-free** セクションにある **firmware-linux** のような別のパッケージに移動されました。このようなパッケージがインストールされていると、必要な時にファームウェアが自動的に読み込まれます。

## パッケージ管理

ターミナルでの話的なパッケージ管理の推奨プログラムは、**aptitude** です。非話的なパッケージ管理用コマンドラインインターフェースとしては、**apt-get** の利用が推奨されます。また、**apt-get** はメジャーリリース間でのアップグレードにも推奨されるツールです。まだ **dselect** を使用しているのであれば、公式のパッケージ管理用フロントエンドとして **aptitude** に移行すべきでしょう。

**squeeze** では、APT は標準設定で自動的に推奨パッケージ (**Recommends**) をインストールします。これは、以下の行を `/etc/apt/apt.conf` に追加することで更できます:

```
APT::Install-Recommends "false";
```

## 依存関係ベースの起動

Debian GNU/Linux の起動システムでの重要な改良点は、依存関係ベースの起動シーケンスと並列起動の導入です。この機能は、新規インストールでは標準で有効になっており、可能であれば **lenny** からのアップグレードでも有効になります。

この機能は、**insserv** を使って **sysv-rc** が **init.d** スクリプトで宣言されている依存関係を元にスクリプトを並び替えることによって有効になります。これは、起動システムそのものに加え、ディストリビューションで提供されているパッケージのすべての起動スクリプトを修正する努力を続けたことにより可能になりました。

依存関係ベースの起動シーケンスで起動システムのスクリプトを並列に実行できるようになり、ほとんどの環境では、起動プロセスの速度を改善します。この機能は、新しいシステムやアップグレードの際、可能な時に標準で有効になります。これを無効にするには、

```
CONCURRENCY=none
```

と `/etc/default/rcS` で指定してください。この機能に関するより詳細な情報は、`/usr/share/doc/insserv/README.Debian` で入手できる情報を参照してください。

## キーボード設定の統一

今回の新しいリリースでは、キーボードの設定が統一され、コンソールと **Xorg** サーバが同じ設定を使うようになりました。キーボードの設定は `/etc/default/keyboard` 設定ファイルで定義されるようになり、**Xorg** の設定ファイルで定義されたキーボード設定を上書きします。

**console-setup** パッケージが、両環境下のキーボード設定やコンソールのフォント設定を取り扱うようになりました。**dpkg-reconfigure keyboard-configuration** を実行するか、手動で `/etc/default/keyboard` 設定ファイルを編集すればキーボードレイアウトと関連設定を再構成できます。

## カーネルモードセッティング (kernel mode setting, KMS)

(Intel、ATI/AMD、NVIDIA から出荷されている) もっとも一般的なデスクトップ用チップセットのグラフィックモード設定コードは、それぞれ異なる **Xorg** ドライバから Linux カーネルに移動しました。これにより、以下のような多くの改善をもたらされました:

より安定したサスペンドとレジューム機能  
X 無しでグラフィックデバイスを利用する機能  
より速い仮想コンソールの切り換え  
ネイティブモードのテキストコンソール

より詳細については、グラフィック関連ソフトウェアの更項 や Debian wiki [<http://wiki.debian.org/KernelModesetting>] にあります。

## LDAP サポート

今回の Debian のリリースでは、LDAP を使ったクライアントサイド認証の実装について、複数のオプションを提供しています。`libnss-ldap` パッケージおよび `libpam-ldap` パッケージのユーザは、`libnss-ldapd` パッケージおよび `libpam-ldapd` パッケージへのアップグレードを討 する必要があります。

これらの新しいパッケージは、LDAP クエリを中央の非特 デーモン (`nslcd`) に移譲します。このデーモンは、LDAP 情報を使うプロセスと LDAP クエリを実行するデーモンとに分離をします。これによって、暗号化された LDAP 接続や LDAP 認証のクレデンシャルの取扱い、接続のフェイルオーバーとデバッグ作業を行うためのよりシンプルな仕組みの提供が簡易になり、多くのアプリケーションで LDAP および関連ライブラリの呼び出しを避けることができます。

`libnss-ldapd` および `libpam-ldapd` へのアップグレードは、既存の設定情報がほとんど使いまわせるので簡単です。高度な設定の場合のみ、手動での再構成が必要になります。

しかし、これらのパッケージでは現在のところネストされたグループのサポートがまだ欠けており、LDAP `password modify EXOP` オペレーションを使ったパスワードの更のみをサポートしています。

## proposed-updates セクション

リリース みの安定版 (および古い安定版) ディストリビューションへの更 はすべて、アーカイブに收容される前に長いテスト期間を ています。安定版 (および古い安定版) リリースへのそのようなアップデートはポイントリリースと呼ばれます。ポイントリリースの準備は、`proposed-updates` の仕組みを通して提供されます。

パッケージが `proposed-updates` に入るには2つの路 があります。まず、[security.debian.org](http://security.debian.org) に加わったセキュリティパッチ適用 みのパッケージは、自動的に `proposed-updates` にも加わります。次に、Debian GNU/Linux 開発者が安定版リリースマネージャのレビューを受けた後で `proposed-updates` へ新しいパッケージを直接アップロードすることがあります。現在のパッケージ一覧は <http://ftp-master.debian.org/proposed-updates.html> で参照できます。

パッケージへのアップデートがポイントリリースに加わる前にテストするのを手伝ってくださる方は、`proposed-updates` セクションを `sources.list` に加えることでご協力いただけます:

```
deb      http://mirrors.kernel.org/debian squeeze-proposed-updates
main contrib
deb-src  http://mirrors.kernel.org/debian squeeze-proposed-updates
main contrib
```

`apt-get update` コマンドを次回実行した時に、システムは `proposed-updates` セクションに含まれるパッケージを認識し、アップグレードするパッケージを探すときにそれらを考慮するようになります。

これは密には Debian の新機能ではありませんが、以前はあまり注目されていなかった機能ではあります。

## stable-updates セクション

`proposed-updates` のパッケージのいくつかは `squeeze-updates` 機構 由 でも利用可能になります。この仕組みは、ウィルススキャナのデータやタイムゾーンのデータなど、次のポイントリリースが行われる前にシステムへインストールしたいと思うであろう更新に使われます。`squeeze-updates` の全パッケージはポイントリリースに含まれません。

これは、以前に `volatile.debian.org` アーカイブ [<http://volatile.debian.org/>] で提供していた機能を置き換えるものであることに注意ください。

`squeeze-updates` からのパッケージを使うには、`sources.list` にエントリを付け加えます:

```
deb      http://mirrors.kernel.org/debian squeeze-updates main
contrib
deb-src  http://mirrors.kernel.org/debian squeeze-updates main
contrib
```

`apt-get update` コマンドを次回実行した時に、システムは `>squeeze-updates` セクションに含まれるパッケージを認識し、アップグレードするパッケージを探すときに処理するようになります。

新しいパッケージが `squeeze-updates` 由 で利用可能になると、`debian-stable-announce` [<http://lists.debian.org/debian-stable-announce/>] メーリングリストで告知されます。

## backports.org/backports.debian.org

`backports.org` リポジトリによって提供されていたサービスは Debian のインフラストラクチャに統合され、現在は公式な Debian のサービスとなっており [<http://www.debian.org/News/2010/20100905>]、`backports.debian.org` [<http://backports.debian.org/>] で提供されています。

## Debian Live

`squeeze` において、Debian は amd64 および i386 アーキテクチャ用の公式な Live システムを提供します。

Debian Live システムは、インストールする必要がなく、リムーバブルメディア (CD-ROM、DVD、USB キー) から直接起動できるか、またはネットワークを介して他のコンピュータから起動できる Debian システムのことです。そのイメージは `live-build` というツールによって生成されたものです。これを利用すると、カスタムメイドのライブイメージを簡単に作れます。Debian Live プロジェクトについての詳細は、<http://live.debian.net/> を参照してください。

## ニューロイメージング研究の包括的なサポート

Debian GNU/Linux 6.0 は、磁気共鳴画像 (Magnetic Resonance Imaging、MRI) をベースにしたニューロイメージング研究の包括的なサポートを提供する初めての GNU/Linux ディストリビューションです。これには構造的画像解析 (例: `ants`)、散強調画像および繊維追跡 (例: `mrtrix`)、刺激伝達 (例: `psychopy`)、MRI のシーケンス開発 (例: `odin`)、多数の多目的データ処理・分析スイート (例: `nipype`) 用の最新のソフトウェアが含まれています。さらに、今回のリリースではすべてのメジャーなニューロイメージングデータ形式のサポートを内 しています。含まれるソフトウェアの全一覧は Debian Science [<http://blends.alioth.debian.org/science/tasks/neuroscience-cognitive>] および Debian Med [<http://debian-med.alioth.debian.org/tasks/imaging>] のタスクページを、そして詳細な情報については NeuroDebian のウェブページ [<http://neuro.debian.net>] を参照してください。

## 第3章 インストールシステム

Debian Installer は公式の Debian インストールシステムです。このインストーラは、様々なインストール方法を提供しています。お使いのシステムにインストールするのにどの方法が利用できるかは、使っているアーキテクチャに依存します。

squeeze 用のインストーラのイメージは、インストールガイドとともに Debian のウェブサイト [<http://www.debian.org/releases/stable/debian-installer/>]にあります。

インストールガイドは、Debian 公式 CD/DVD セットの 1 枚目の CD/DVD 内の、次の場所にも含まれています。

`/doc/install/manual/言語/index.html`

これまでに知られている問題点を列した `debian-installer` の正誤表 [<http://www.debian.org/releases/stable/debian-installer/index#errata>]も確認しておくといでしょう。

### インストールシステムの最新情報

Debian Installer は、Debian GNU/Linux 3.1 (「sarge」)での初めての公式リリース以降も活発に開発されています。その結果、ハードウェアサポートが改良され、ワクワクするような新機能がいくつか追加されました。

このリリースノートでは、インストーラにおける更点のうち主要なもののみをリストアップします。lenny 以降になされた詳細な更一覧の概要に興味がある場合は、Debian Installer のニュースの履歴 [<http://www.debian.org/devel/debian-installer/News/>]で覧可能な、squeeze 用ベータ版やリリース候補版 (RC) のリリースアナウンスを参照してください。

## 主要な更点

サポート外となったプラットフォーム	<p>インストーラでの Alpha ('alpha')、ARM ('arm')、そして HP PA-RISC ('hppa') アーキテクチャへのサポートは行われなくなりました。'arm' アーキテクチャは <b>armel</b> 移植版によって不要となったためです。</p>
kFreeBSD のサポート	<p>インストーラが <b>Linux</b> カーネルではなく <b>kFreeBSD</b> をインストールに使えるようになり、テクノロジープレビューを体験できます。この機能を使うには、適切なインストールイメージ (あるいは CD/DVD のセット) を使う必要があります。</p>
GRUB 2 が標準のブートローダ インストール作業中のヘルプ表示	<p>標準でインストールされるブートローダは <b>grub-pc (GRUB2)</b> です。</p> <p>インストール作業中のダイアログがヘルプ情報を表示するようになりました。現在のところ、すべてのダイアログで使われてはいませんが、この機能は今後のリリースで追加されていきます。これは、特に新規ユーザにして、インストール作業中の利用感を改善してくれます。</p>
推奨パッケージ (Recommends) のインストール	<p>インストールシステムは、一般的な設定が不都合な状況を招く特定の状況下を除いて、インストール作業全体を通して標準で推奨 (Recommends) のパッケージをすべてインストールするようになりました。</p>
ハードウェア固有パッケージの自動インストール	<p>システムは、適切な時にハードウェア固有のインストール作業を自動的に選 びます。これは、<b>discover</b> パッケージの <b>discover-pkginstall</b> の利用によって達成されました。</p>
以前のインストールのサポート	<p>インストールシステムは、<b>lenny</b> のような以前のリリースのインストールにも使えるようになっています。</p>
ミラー選 びの改良	<p>インストールシステムは、<b>lenny</b> 同 じに <b>squeeze</b> や以前のリリースについて、より改善されたサポートを提供します (<b>archive.debian.org</b> を利用します)。それに加え、選 ばれたミラーの一貫性をチェックし、選 ばれたリリースを保持します。</p>
パーティション機能の更 新	<p>今回のインストーラのリリースでは、<b>ext4</b> ファイルシステムの利用がサポートされており、<b>RAID</b>、<b>LVM</b>、暗号で保護されたパーティションシステムの作成が簡易化されています。<b>reiserfs</b> ファイルシステムのサポートは、標準ではもはや含まれていませんが、追加で読み込むことは可能です。</p>
インストール中のファームウェアパッケージの読み込みサポート	<p>リムーバブルメディアに加えて、インストールメディアからのファームウェアパッケージファイルの読み込みが可能になり、これによってファームウェアを含めた <b>PXE</b> イメージと <b>CD/DVD</b> の作成ができるようになりました。</p>
	<p><b>Debian 6.0</b> から、<b>non-free</b> のファームウェアは <b>main</b> から外されました。<b>non-free</b> なファームウェアを必要とするハードウェアで <b>Debian</b> をインストールするには、インストール中にファームウェアを自分で用意するか、ファームウェアを含んでいる <b>non-free</b> な作成 された <b>CD/DVD</b> を使うかしてください。詳細については、<b>Debian</b> のウェブサイト上の <b>Debian</b> を入手するには [<a href="http://www.debian.org/distrib">http://www.debian.org/distrib</a>] を参照してください。</p>
新しい言語	<p>翻訳者による多大な努力によって、<b>Debian GNU/Linux</b> は 67 の言語でインストールできるようになりました。これは <b>lenny</b> より三つの言語が追加されています。ほとんどの言語はテキスト版ユーザインターフェイスでのインストールとグラフィカルユーザインターフェイスでのインストールの双方が可能になっていますが、いくつかはグラフィカルユーザインターフェイスでのみ利用が可能です。</p>

このリリースで追加された言語:

アストゥリアス語、エストニア語、カザフ語、ペルシア語がグラフィカルインストーラとテキスト版インストーラに追加されました。

カンナダ語とテルグ語がグラフィカルインストーラに追加されました。

タイ語は、以前はグラフィカルユーザインターフェイスのみで利用可能でしたが、テキスト版インストーラのユーザインターフェイスでも利用可能になりました。

## 自動インストール

前のセクションで述べたとおりインストーラに多数の更 改 が加えられたので、インストーラによる、事前設定ファイルを使用した自動インストールのサポートにも更 改 がありました。つまり、**lenny** のインストーラで動いた既存の事前設定ファイルがあっても、修正を加えずにそれが新しいインストーラで動くことは期待できません。

事前設定の使用方法に関する豊富な文書を含んだ最新の付 録 が、インストールガイド [<http://www.debian.org/releases/stable/installmanual>]に追加されました。

# 第4章 Debian 5.0 (lenny) からのアップグレード

## アップグレードの準備

アップグレードの前には、第5章に書かれている情報も読むことをお勧めします。この章に書かれている問題点は、アップグレードの過程と直接は関係がないかもしれませんが、それでもアップグレードを開始する前に知っておくべき重要事項である可能性があります。

### あらゆるデータや設定情報をバックアップする

システムをアップグレードする前に、完全なバックアップを取っておくよう強くお勧めします。少なくとも、失いたくないデータや設定情報だけでもバックアップしておきましょう。アップグレードのツールや処理はきわめて信頼性の高いものですが、アップグレードの最中にハードウェア障害が起こると、システムに大きなダメージを与えることがあります。

バックアップしておくべき主な象として、`/etc`、`/var/lib/dpkg`、`/var/lib/apt/extended_states` の中身、`dpkg --get-selections "*" (引用符を忘れてはいけません)` の出力などがあります。システムの管理に **aptitude** を使っている場合は、`/var/lib/aptitude/pkgstates` もバックアップしておくといいでしょう。

アップグレードの過程自体は、`/home` ディレクトリ以下は一切更しません。とはいえ、(Mozilla スイートの一部や GNOME や KDE のデスクトップ環境などのように) ユーザが初めて新しいバージョンのアプリケーションを起動するときに、既存のユーザ設定を新たなデフォルト値で書き替えてしまうものがあるのも事実です。万一に備えて、ユーザのホームディレクトリにある `.sh` ファイルと `.sh` ディレクトリ (いわゆる「ドットファイル」) をバックアップしておくのがよいでしょう。古い状態にしたり、再度設定する場合に役立つはずですが、ユーザにもこのことについて知らせておいてください。

あらゆるパッケージのインストール処理はスーパーユーザ特 で実行されなければならないため、`root` としてログインするか `su` や `sudo` を使って、必要なアクセス限 を得てください。

アップグレードにあたって事前に整えなければならない条件がいくつかあります。実際にアップグレードを実行する前にそれらを確認してください。

### 事前にユーザに通知する

アップグレードの前には、その予定をすべてのユーザに知らせるとよいでしょう。ただ、システムに **ssh** 接続などでアクセスしてきているユーザが、アップグレードの最中にそうと気付くことはほとんどないはずで、また、作業を続行できるはずですが。

万一の策 をしたければ、アップグレードの前に `/home` パーティションをバックアップするか、アンマウントしてしまいましょう。

**squeeze** にアップグレードするときはおそらくカーネルをアップグレードしなければならないので、通常は再起動が必要です。

### サービスのダウン期間の準備

アップグレード作業を行う場合、この作業に含まれるパッケージに関連しているサービスがあることと思います。この場合、アップグレードされるパッケージが置き換えられて設定される間、サービスが停止するかもしれません。この間、サービスが利用できなくなります。

これらのサービスに する実際のダウン期間は、システム中でアップグレードされるパッケージ数に じて違いますし、このダウン期間には (もしあれば) 各パッケージのアップグレードに する設定の質問にシステム管理者が答える時間も含まれます。アップグレード作業が放置されたまままでいて、システムがアップグレード中に入力を必要とした場合、非常に長期間 サービスが利用できなくなる可能性が高いことにご注意ください。

アップグレードされるシステムが、ユーザあるいはネットワークに して非常に重要な サービスを提供している場合、システムの最小アップグレード項 で記述しているように最 小限のシステムアップグレードを行い、次にカーネルのアップグレードと再起動 (カーネ ルと **udev** のアップグレード項 参照) をし、そしてもっとも重要なサービスに関連する パッケージをアップグレードすれば、ダウン期間を減らすことができます。

## 復旧の準備

ドライバやハードウェア 出、 デバイスファイルの命名法や順序に関して **lenny** と **squeeze** との間ではカーネルに多くの 更 が加えられたため、アップグレード後のシステ ム再起動で問題に直面するリスクが高くなっています。既知の潜在的な問題点の多くは、 このリリースノートの本章と次章で述べられています。

上述の理由により、システムが再起動に失敗したり、リモート管理されているシステム ならネットワーク接続の確立に失敗した場合に備え、復旧できる手立てを整えておくこと が大切です。

**ssh** 接続 由 でリモートからアップグレードを行うのなら、リモートのシリアル端末から サーバにアクセスできるよう、必要な事前準備をしておくことを強くお勧めします。カー ネルをアップグレードして再起動した後、(デバイスの整列順序の 更項 で述べるように) 一部のデバイス名が 更 され、ローカルコンソール 由 でシステム設定を修正しなければ ならないことがあります。また、アップグレード中に誤ってシステムが再起動された場合 にも、ローカルコンソールを使って復旧する必要に迫られることがあります。

最初に試すべきもっとも明白なことは、古いカーネルでの再起動です。しかしながら、 本文書の別の場所で述べられているいくつかの理由により、これがうまくいくという保証 はありません。

古いカーネルでの再起動に失敗するなら、システムを起動してアクセス・修復するた めの代替手段が必要となるでしょう。1つのオプションとしては、特別な復旧イメージや **Linux** ライブ CD を使うことがあります。これらを使って起動した後は、ルートファイ ルシステムをマウントし、**chroot** でその中に入って問題点を調・解決 できるはずで す。

お勧めしたい別のオプションとしては、**squeeze** 用 **Debian Installer** のレスキュー モードを使用する方法があります。同インストーラを使う利点は、多くのインストール手 段の中からあなたの状況に最適なものを選べることです。より詳しい情報は、インストー ルガイド [<http://www.debian.org/releases/stable/installmanual>]の第 8 章にある 「壊れたシステムの復旧」 セクションや、**Debian Installer FAQ** [[http:// wiki.debian.org/DebianInstaller/FAQ](http://wiki.debian.org/DebianInstaller/FAQ)]を参照してください。

## **initrd** を使った起動中のデバッグシェル

**initramfs-tools** で生成される **initrd** には、デバッグシェル が含まれています。例 えば、**initrd** がルートファイルシステムをマウントできなければ、デバッグシェル内に移 るでしょう。このデバッグシェルは、問題の追跡、そしておそらくは修正の手助けとなる 基本的なコマンドを備えています。

チェックすべき基本的事項としては、次のようなものがあります。**/dev** 内に適切なデバ イスファイルが存在するか、どのモジュールがロードされているか (**cat /proc/ modules**)、**dmesg** の出力にドライバのロード失敗のエラーが出ていないか、など。 **dmesg** の出力はまた、どのデバイスファイルがどのディスクに割り当てられているのか も示してくれます。ルートファイルシステムが期待通りのデバイス上にあるかを確認す るために、**echo \$ROOT** の出力もチェックすべきでしょう。

問題点を何とか解決できたなら、**exit** とタイプすることでデバッグシェルを終了させ、 起動プロセスを失敗した時点から 続 できます。もちろん次回の起動時に再び失敗するこ とが無いよう、根本的な問題を修正して **initrd** を再生成する必要があるでしょう。

## アップグレード用の安全な環境の準備

ディストリビューションのアップグレードは、ローカルのテキストモード 仮想コンソール (あるいは直接接続されたシリアル端末) から行うか、リモートなら **ssh** 接続 由 で行 いましょう。

### 重要項目

(**tinc** のような)VPN サービスを使っている場合、アップグレード作業中に使えなくなるかもしれません。サービスのダウン期間の準備項を参照してください。

リモートでのアップグレード時にさらなる安全性のマーヅンを得るため、**screen** プログラムが提供する仮想コンソール内でアップグレード作業を行うことを提案します。同プログラムは安全な再接続を可能にし、リモート接続プロセスが切断された場合でもアップグレード作業が中断しないようにしてくれます。

## 重要項目

**telnet**、**rlogin**、**rsh** を用いてアップグレードをしてはいけません。アップグレードするマシンの **xdm**、**gdm**、**kdm** などが管理している X セッションからのアップグレードも行うべきではありません。これらのサービスはアップグレードの最中に切断されてしまう可能性が高く、そうするとアップグレード途中のシステムへの接続が不可能になってしまうからです。

## 衝突しているパッケージを削除する

bug #512951 [<http://bugs.debian.org/512951>] のため、**splashy** パッケージをアップグレードの前に完全に削除 (**purge**) しておく必要があります。

```
# apt-get purge splashy
```

## システムの状態をチェックする

この章で述べられているアップグレード手順は、サードパーティ製のパッケージが無い「純粋」な **lenny** システムからのアップグレード用です。アップグレードプロセスにおいて最大限の信頼性を確保するために、アップグレード開始前にシステムからサードパーティ製パッケージを削除しておいた方が良いでしょう。

5.0 (**lenny**) より古い **Debian** のリリースバージョンからの直接のアップグレードはサポートされていません。5.0 へのアップグレードは、まず **Debian GNU/Linux 5.0** のリリースノート [<http://www.debian.org/releases/lenny/releasenotes>] の指示に従ってください。

またこの手順は、システムが **lenny** の最新ポイントリリースにアップデート済みであるものと想定しています。そうではなかったり、アップグレードみかどうか不明なら、**lenny** システムのアップグレード項内の指示に従ってください。

## パッケージマネージャにおいて中断しているアクションの確認

パッケージをインストールするのに **aptitude** の代わりに **apt-get** を使用すると、時として、**aptitude** がそのパッケージを「使われていない」とみなし、削除象とすることがあります。一般的には、アップグレードを先に進める前に、システムが完全に最新かつ「クリーン」な状態となっているかを確認する必要があります。

そのため、パッケージマネージャ **aptitude** において中断しているアクションがないか確認すべきでしょう。パッケージマネージャにおいて、あるパッケージが削除あるいは更新の象となっているなら、アップグレード手順に好ましくない影響を与えるかもしれません。パッケージマネージャにおけるアクションの修正は、**sources.list** に **stable** や **squeeze** ではなく、**lenny** が指定されている段階でのみ可能なことに注意してください。ソースリストのチェック項も参照してください。

この確認を行うには、**aptitude** を「ビジュアルモード」で起動して **g** (「Go」) を押してください。何らかのアクションが表示されたなら、その内容を確認して、修正するかあるいは提案されたアクションを実行すべきです。いかなるアクションも提案されない場合は、「インストール・削除・更新予定のパッケージがありません」というメッセージが表示されるでしょう。

## APT の pin 機能を無効にする

特定のパッケージを安定版以外のディストリビューション (テスト版など) からインストールするように APT を設定しているなら、当該パッケージが新しい安定版リリース内のバージョンにアップグレードできるように、(/etc/apt/preferences 内に保存されている) APT の pin 設定を変更する必要があるかもしれません。APT の pin 機能に関するより詳しい情報は、apt\_preferences(5) にあります。

## パッケージの状態をチェックする

アップグレードに使う手段に関係なく、まず全パッケージの状態を調べ、全パッケージがアップグレード可能な状態にあるのを確認することをお勧めします。次のコマンドは、インストールが未完了のパッケージ (Half-Installed) や設定に失敗したパッケージ (Failed-Config)、何らかのエラー状態にあるパッケージを表示します:

```
# dpkg --audit
```

**dselect** や **aptitude**、あるいは次のようなコマンドを使ってシステムの全パッケージの状態を確認することもできます:

```
# dpkg -l | pager
```

または

```
# dpkg --get-selections "*" > ~/curr-pkgs.txt
```

アップグレード前に、あらゆる **hold** 状態を解除しておいたほうがよいでしょう。アップグレードに不可欠なパッケージが **hold** 状態にあるなら、アップグレードに失敗するでしょう。

**hold** 状態にあるパッケージを記すのに、**aptitude** は **apt-get** や **dselect** とは異なる手法を用いることに注意してください。**aptitude** で **hold** 状態にあるパッケージを確認するには、以下のように実行します:

```
# aptitude search "~ahold" | grep "^h"
```

**apt-get** でどのパッケージが **hold** 状態にあるのかを調べたければ、以下のように実行してください:

```
# dpkg --get-selections | grep hold
```

パッケージをローカルで更したり再コンパイルしており、パッケージの名前をえたりバージョン番号に **epoch** フィールドを追加していないなら、アップグレードしないよう **hold** 状態にしておかなければなりません。

**apt-get** でパッケージを「**hold**」状態に変更するには、以下のように実行してください:

```
# echo パッケージ名 hold | dpkg --set-selections
```

「**hold**」状態を解除するには **hold** の代わりに **install** を使用してください。

修正が必要なことがあるなら、ソースリストのチェック項で明 するように `sources.list` が `lenny` を指定したままにしておくべきです。

## proposed-updates セクション

`/etc/apt/sources.list` ファイルに `proposed-updates` セクションを含めている場合は、システムのアップグレードを試みる前に、それらのセクションをファイルから削除してください。これは、衝突の可能性を減らすための予防策です。

## 非公式なソースとバックポートパッケージ

システムに `Debian` 以外のパッケージがインストールされている場合、依存関係の衝突のためアップグレード中に削除されるかもしれないことに注意すべきです。当該パッケージが `/etc/apt/sources.list` に `Debian` 以外のパッケージアーカイブを追加することでインストールされたのなら、そのアーカイブが `squeeze` 用にコンパイルされたパッケージも提供しているかをチェックし、`Debian` パッケージ用のソース行と一緒にそのソース行も適切に修正すべきです。

非公式にバックポートされた、`Debian` に存在するパッケージの「新バージョン」を `lenny` システムにインストールしているユーザもいるかもしれません。そのようなパッケージはファイルの競合に繋がる可能性があるため、アップグレード中に問題を引き起こす場合がほとんどでしょう。ファイルの競合が発生した場合の処方法 については、アップグレード中の注意点項にいくつかの情報があります。

## APT の取得先 (ソース) の準備

アップグレードを始める前に、`apt` の設定ファイル `/etc/apt/sources.list` を編集して、パッケージ一覧の取得先を設定する必要があります。

`apt` は、あらゆる「`deb`」行を通して見つかったすべてのパッケージを見比べ、最も大きなバージョン番号のパッケージをインストールします。同じパッケージが取得可能な場合は、ファイルで最初に現れた行を優先します (したがって、複数のミラーを指定する場合は、最初にローカルのハードディスクを、次に `CD-ROM` を、最後に `HTTP/FTP` ミラーを指定するといいでしょう)。

リリースを指定するのに、コードネーム (`lenny` や `squeeze`) と状態名 (`oldstable`、`stable`、`testing`、`unstable`) のどちらもよく使用されます。コードネームによる指定には、新しいリリースが出たときに驚かずに むとという利点があるため、ここではコードネームを使用しています。当然ですが、コードネームを使用している場合は自分でリリースアナウンスに注意を払わなければいけません。代わりに状態名を使用している場合は、リリースが行われた直後に、パッケージが大量に更新可能になったことに気づくでしょう。

## APT のインターネットソースの追加

デフォルトの設定では、メインの Debian インターネットサーバを使ってインストールするようになっています。ですが、`/etc/apt/sources.list` を編集して、他のミラー (できればネットワーク的に最も近いミラー) を使うようにするほうがよいでしょう。

Debian の HTTP/FTP ミラーのアドレスは、`http://www.debian.org/distrib/ftplist` にあります (「Debian ミラーサイト一覧」のセクションを参照してください)。一般には HTTP ミラーのほうが FTP ミラーよりも高速です。

例えば、一番近くにある Debian ミラーが `http://mirrors.kernel.org/` だったとしましょう。このミラーをウェブブラウザや FTP プログラムで見ると、主なディレクトリが以下のような構成になっていることがわかります。

```
http://mirrors.kernel.org/debian/dists/squeeze/main/binary-i386/...
http://mirrors.kernel.org/debian/dists/squeeze/contrib/binary-i386/...
```

このミラーを `apt` で使うには、次の行を `sources.list` ファイルに追加します。

```
deb http://mirrors.kernel.org/debian squeeze main contrib
```

新しいソースを追加した後、`sources.list` 内の既存の「`deb`」行の先頭にシャープ記号 (`#`) を追加して、それらを無効にしてください。

## APT のローカルミラーソースの追加

HTTP や FTP のパッケージミラーを使うのではなく、ローカルディスク (おそらくは NFS マウントされたもの) にあるミラーを使うよう、`/etc/apt/sources.list` を更改したいことがあるかもしれません。

例えばパッケージのミラーが `/var/ftp/debian/` にあり、主なディレクトリの配置が次のようになっているとします。

```
/var/ftp/debian/dists/squeeze/main/binary-i386/...
/var/ftp/debian/dists/squeeze/contrib/binary-i386/...
```

これを `apt` で使うには、次の行を `sources.list` ファイルに追加します。

```
deb file:/var/ftp/debian squeeze main contrib
```

`'dists'` は書かなくても暗 のうちに追加されます。リリース名の後の各引数は、パスの末尾につけて、複数のディレクトリとするのに用いられます。

新しいソースを追加した後、`sources.list` 内の既存の「`deb`」行の先頭にシャープ記号 (`#`) を追加して、それらを無効にしてください。

## APT の CD-ROM/DVD ソースの追加

CD だけでインストールをしたい場合は、`/etc/apt/sources.list` 内の「`deb`」行の先頭にシャープ記号 (`#`) を置き、それらを無効にしてください。

CD-ROM ドライブをマウントポイント `/cdrom` にマウントできるようにしている行が `/etc/fstab` にあるかどうかを確認してください (`apt-cdrom` を使う場合は、マウントポイントを `/cdrom` 以外にはできません)。例えば `/dev/hdc` が CD-ROM ドライブなら、`/etc/fstab` には次のような行が必要です。

```
/dev/hdc /cdrom auto defaults,noauto,ro 0 0
```

第 4 フィールドの `defaults,noauto,ro` の単語の間には、スペースを入れてはいけません。

これが正しく機能しているか調べるには、CD を入 して以下を実行してみてください。

```
# mount /cdrom      # マウントポイントに CD をマウントします
# ls -alF /cdrom    # CD のルートディレクトリを表示します
# umount /cdrom     # CD をアンマウントします
```

問題がなければ

```
# apt-cdrom add
```

を、Debian Binary CD-ROM それぞれに して実行してください。各 CD に関するデータが APT のデータベースに追加されます。

## パッケージのアップグレード

Debian GNU/Linux の前回のリリースからのお勧めのアップグレード方法は、パッケージ管理ツール **apt-get** を用いる方法です。前回のリリースでは、この作業には **aptitude** が推 されていましたが、**apt-get** の最近のバージョンでは同等の機能が提供されており、さらにより高い整合性により望ましいアップグレード結果をもたらします。

まず、必要なすべてのパーティション (特にルートパーティションと `/usr` パーティション) を **read-write** モードでマウントするのを忘れずに行いましょう。それには以下のようなコマンドを使います。

```
# mount -o remount,rw /マウントポイント
```

次に、(`/etc/apt/sources.list` 内の) APT ソースのエントリが 「**squeeze**」 と 「**stable**」 のいずれか一方を指定していることを念入りにチェックしてください。**lenny** を指し示すソースエントリが含まれないようにすべきです。

### 注意

CD-ROM のソース行は 「**unstable**」 を指定していることがよくあります。これは混乱の元かもしれませんが、更 すべきではありません。

## セッションの記

ここで強くお勧めしたいのですが、**/usr/bin/script** プログラムを使って、このアップグレードセッションの記を取るようにしましょう。こうすれば、何らかの問題が生じたときに何が起こったかを記しておくことができ、必要に応じてバグ報告に正確な情報を含めることができます。記を開始するには次のように入力します。

```
# script -t 2>~/upgrade-squeeze.time -a ~/upgrade-squeeze.script
```

**typescript** ファイルは **/tmp** や **/var/tmp** のような一時ディレクトリには置かないでください (これらのディレクトリ内のファイルはアップグレードや再起動の際に削除されることがありますから)。

アップグレード完了後に **script** を停止するには、プロンプトから **exit** と入力してください。

**script** に **-t** スイッチをつけておいた場合は、以下のように **scriptreplay** プログラムでセッション全体をリプレイできます。

```
# scriptreplay ~/upgrade-squeeze.time ~/upgrade-squeeze.script
```

## パッケージリストの更新

まず、新しいリリースで利用可能なパッケージの一覧を取得する必要があります。そのためには以下のコマンドを実行してください。

```
# apt-get update
```

## アップグレードするのに十分な領域があることを確認する

システムアップグレードの前には、システムのアップグレード項で明するシステム全体のアップグレードを開始するときに十分なハードディスク領域があるかどうかを確認しなければいけません。まず、ネットワーク由で取得してインストールする必要があるどのようなパッケージも、`/var/cache/apt/archives` (およびダウンロード中には `partial/` サブディレクトリ) に保存されます。したがって、システムにインストールされるパッケージをダウンロードして一時的に保存できるよう、`/var/` を保持しているファイルシステムパーティションに十分な空き領域があることを確認しなければなりません。ダウンロード後にはおそらく、アップグレードされるパッケージ (これらには、より大きなバイナリやより多くのデータが含まれている可能性があります) と、アップグレードに伴って依存関係に引きずられて新たにインストールされるパッケージの両方のインストールのために、他のファイルシステムパーティションにさらに領域が必要になるでしょう。システムに十分な空き領域がない場合、アップグレードが不完全な状態で終わり、復旧が困難になる可能性があります。

**apt-get** で、インストールに必要なディスク領域の詳細な情報が表示できます。アップグレードを実行する前に、次のように実行して必要な領域の推定値を見ることができます。

```
# apt-get -o APT::Get::Trivial-Only=true dist-upgrade
```

```
[ ... ]
```

更新: XXX 個、新規インストール: XXX 個、削除: XXX 個、保留: XXX 個。

アーカイブ yyyMB 中 xx.xMB を取得する必要があります。展開後に AAAMB のディスク

領域が新たに消費されます。

パッケージのダウンロード・インストール・削除をおこないます。

アップグレード手順の初めにこのコマンドを実行すると、以降のセクションで明するような理由でエラーが発生する可能性があります。その場合は、このコマンドを実行してディスク領域の推定値を見る前に、まずシステムの最小アップグレード項で明するとおりシステムの最小アップグレードを行い、さらにカーネルをアップグレードする必要があります。

アップグレードをするのに十分な領域がない場合は、**apt-get** が以下のような警告メッセージを出します:

エラー: `/var/cache/apt/archives/` に十分な空きスペースがありません。

この場合、事前に領域を解放するのを忘れないようにしてください。以下のことを実行するとよいでしょう。

インストールのために以前 (`/var/cache/apt/archives` に) ダウンロードしたパッケージを削除する。**apt-get clean** を実行してパッケージキャッシュを一掃すると、以前ダウンロードしたパッケージファイルをすべて削除できます。

忘れ去られたパッケージを削除する。**popularity-contest** をインストールしていただければ、**popcon-largest-unused** を使って、使用していないパッケージのうち最も大きな領域を占めているものをリストアップできます。**deborphan** や **debfooster** を使って時代別のパッケージを見つけることも可能です (時代別 (Obsolete) のパッケージ項を参照してください)。それらのツールを使う代わりに **aptitude** を「ビジュアルモード」で起動すれば、古いパッケージは、「止された、またはローカルで作成されたパッケージ」の下に見つかります。

あまりにも大きな領域を占めており現在は必要ないパッケージを削除する (アップグレード後にいつでも再インストール可能なのですから)。**dpigs** (`debian-goodies` パッケージに含まれています) や **wajig** (`wajig size` を実行してください) を用いると、最も大きなディスク領域を占めているパッケージをリストアップできます。

**aptitude** を使っても、ディスク領域の大部分を占めているパッケージをリストアップできます。**aptitude** をビジュアルモードで起動します。表示 → 平坦なパッケージ一覧を新規に作成を実行し、**l**を押して、**~i**と入力します。**S**を押して **~installsize**と入力します。すると、作業しやすい一覧が得られます。

パッケージを安全に削除するための注意として、ソースリストのチェック項で明らかに、**sources.list**が **lenny**を指し示すよう設定をしておくことが望ましいです。

## システムの最小アップグレード

完全アップグレード (以下に記述しています) を直接行った場合、残しておきたいパッケージが大量に削除されてしまうことが時折あります。そのため、まずはこれらの競合状態を打開するための最小アップグレードを行い、その上で システムのアップグレード項にあるような完全な **dist-upgrade**を行う、という 2段階のアップグレード過程を踏むことをお勧めします。

これをまず行うには、以下のコマンドを実行してください:

```
# apt-get upgrade
```

このコマンドには、アップグレードしても他のパッケージをインストール・削除する必要がないパッケージだけをアップグレードする、という効果があります。

システムの容量が少なく、容量による制約のため完全アップグレードが実行できない場合にも、システムの最小アップグレードは有用です。

## カーネルと **udev** のアップグレード

squeeze のバージョンの udev は、CONFIG\_SYSFS\_DEPRECATED が無効に、そして CONFIG\_INOTIFY\_USER と CONFIG\_SIGNALFD オプションが有効になっている 2.6.26 以降のカーネルを必要とします。lenny での標準 Debian カーネル (バージョン 2.6.26) は CONFIG\_SYSFS\_DEPRECATED が有効になっており、lenny 中のバージョンの udev は最新のカーネルが要求する機能をすべて提供していなかったため、アップグレードの際にはシステムを起動不可能な状態にしないように特別に注意を払う必要があります。

lenny の 2.6.26 カーネルを squeeze の udev と共に起動するとネットワークデバイスに正しく名前を割り当てることや、(disk グループによってアクセスされるような) ブロックデバイスに特定の追加パーミッションを適用するのに失敗することがあります。ソフトウェアそのものは動いている に見えますが、いくつかのルール (例えば、ネットワークベースのルール) が正しく読み込まれません。従って、udev をアップグレードする前に互換性があるカーネルが利用可能になっているのを確実にするため、この時点でカーネルそのものをアップグレードすることを強くお勧めします。

このカーネルアップグレードを実行するには、次のコマンドを実行してください。

```
# apt-get install linux-image-2.6-flavor
```

カーネルパッケージのどのフレーバーをインストールすべきかの判断に手助けが必要な場合は、カーネルメタパッケージのインストール項を参照してください。

grub ブートローダのユーザは、カーネルのアップグレードの一部としてして **update-grub** が走るのを確認するか、手動で実行する必要があります。

カーネルのアップグレード直後に、古い udev と新しいカーネルを使うことによる非互換性のリスクを最小限にするため、新しい udev もインストールする必要があります。これを行うには以下を実行します:

```
# apt-get install udev
```

一旦カーネルと udev の両方をアップグレードし終わったら、システムを再起動する必要があります。

## システムのアップグレード

これまでの手順を実行し終わったら、アップグレードの主要な部分を続ける準備ができています。以下のコマンドを実行してください:

```
# apt-get dist-upgrade
```

### 注意

他のリリースでのアップグレード作業では、アップグレードに **aptitude** の利用を推 していました。このツールは lenny から squeeze へのアップグレードには推 されません。

これによってシステムの完全なアップグレードが行われます。すなわち、すべてのパッケージの最新版がインストールされ、リリース間で発生しうるパッケージの依存関係の化すべてが解決されます。必要に じて、新しいパッケージ (通常は、新しいバージョンのライブラリや、名前の わったパッケージ) がインストールされたり、衝突した古いパッケージが削除されたりもします。

CD-ROM (または DVD) のセットからアップグレードする場合には、アップグレードの最中に、特定の CD を入れるよう何回か指示されることになります。同じ CD を複数回入れなければならないかもしれません。これは、相互に依存しているパッケージが別々の CD に分散しているためです。

現在インストールされているパッケージを新しいバージョンへとアップグレードする際に、他のパッケージのインストール状態を更しななければならないような場合には、そのパッケージは現在のバージョンのままになります（「固定されている」と表示されます）。この状態は、**aptitude** でこれらのパッケージをインストール象として選ぶか、または **apt-get -f install** パッケージ名を実行してみると、解決できます。

## アップグレード中の注意点

以下の章では、**squeeze** へのアップグレードの最中に現れるかもしれない既知の問題を記述しています

## **squeeze** の **Linux** カーネルには **cryptloop** のサポートが含まれていません

**cryptloop** のサポートは、**Debian 6.0** の **Linux** カーネルパッケージから外されています。 **cryptloop** を使っている既存のインストールは、アップグレード前に **dm-crypt** へ移行する必要があります。

## 予期されるパッケージの削除

**squeeze** へのアップグレード作業では、システム中のパッケージ削除を尋ねてくるかもしれません。実際のパッケージ一覧は、インストールしてあるパッケージの構成に従って異なってくるでしょう。このリリースノートでは、どのような方法をとるべきかに関する一般的なアドバイスをします。しかし、確信がもてない場合は、それぞれの方法でアップグレードを先に進める前に、どのパッケージを削除するよう提案されているのか、きちんと調べることをお勧めします。

削除されるだろうと予想されるパッケージには以下が含まれています: **autofs** (**autofs5** で置き換え)、**dhcp3** (**isc-dhcp** で置き換え)、**madwifi-source**、**python-setuptools** および **python2.4** (**python2.6** で置き換え)。**squeeze** で時代れとなるパッケージについての詳細な情報は、時代れ (**Obsolete**) のパッケージ項を参照してください。

## **aptitude** あるいは **apt-get** を実行中のエラー

**aptitude** や **apt-get**、**dpkg** を使用した操作が次のようなエラーで失敗に終わるかもしれません。

```
E: Dynamic MMap ran out of room
```

この場合、デフォルトのキャッシュ容量では不十分だということになります。これを解決するには、**/etc/apt/sources.list** から不要な行を削除もしくはコメントアウトするか、キャッシュサイズを やします。キャッシュサイズを やすには、**/etc/apt/apt.conf** に **APT::Cache-Limit** を設定します。以下のコマンドを実行すれば、アップグレードするのに十分な値が設定されます:

```
# echo 'APT::Cache-Limit "12500000";' >> /etc/apt/apt.conf
```

ここでは、**/etc/apt/apt.conf** ファイル内にまだこの値を設定していない場合を想定しています。

## 衝突 (**Conflicts**) あるいは先行依存 (**Pre-Depends**) のループ

場合によっては衝突や事前依存のループのために、**APT** の **APT::Force-LoopBreak** オプションを有効にして、必須パッケージを一時的に削除しなければならないかもしれません。その場合 **apt-get** はこのことを警告してアップグレードを中断します。**apt-get** のコマンドラインにオプション **-o APT::Force-LoopBreak=1** を指定すれば、この状態を回避できます。

システムの依存関係の構造があまりに問題だらけで、手動での介入が必要となることもあります。通常、手動での介入とは、**apt-get** を用いるか、あるいは

```
# dpkg --remove パッケージ名
```

で問題の原因となるパッケージを消す作業になります。または次の方法を用いてもよいかもしれません。

```
# apt-get -f install  
# dpkg --configure --pending
```

極端な場合には、コマンドラインから次のように入力して、再インストールしなければならないかもしれません。

```
# dpkg --install /path/to/パッケージ名.deb
```

## ファイルの衝突

「純粹」な **lenny** システムからのアップグレードでは、ファイルの衝突は起こらないはずですが、非公式のバックポートパッケージをインストールしているなら起こるかもしれません。ファイルの競合が起こると、次のようなエラーになります:

```
(<package-foo-file> から) <package-foo> を展開しています...  
dpkg: <package-foo> の処理中にエラーが発生しました (--install):  
`<some-file-name>' を上書きしようとしています。これはパッケージ <package-  
bar>  
にも含まれています  
dpkg-deb: サブプロセス paste がシグナル (Broken pipe) によって強制終了しま  
した  
以下のパッケージの処理中にエラーが発生しました:  
<package-foo>
```

ファイルの衝突を解消するには、エラーメッセージの最後の行に表示されたパッケージを強制的に削除します:

```
# dpkg -r --force-depends パッケージ名
```

問題が修正できたら、先程明した **apt-get** のコマンドを再度入力すれば、アップグレードを再開できます。

## 設定の更

アップグレードの最中に、いくつかのパッケージの設定・再設定に関する質問が表示されます。**/etc/init.d** ディレクトリと **/etc/manpath.config** ファイルに関しては、パッケージメンテナのバージョンに置き換えるようにしてください。システムの整合性を保つためには **'yes'** と答えることが必要になります。古いバージョンも **.dpkg-old** という張子をつけられて保存されていますので、すのはいつでもできます。

どうすればよいかわからなくなったら、そのパッケージやファイルの名前を書き留めておいて、その問題解決は後回しにしましょう。**typescript** ファイルを索すれば、アップグレードの最中に画面に表示された情報をもう一度見ることもできます。

## コンソール接続へセッションの更

システムのローカルコンソールを使ってアップグレードを実行している場合、アップグレードの最中に何回かコンソールが別の画面へ移動してしまい、アップグレード作業が見えなくなることに気づくかもしれません。例えば、デスクトップシステムで **gdm** が再起動した際に起こります。

仮想ターミナル 1 には **Ctrl+Alt+F1**、あるいはローカルのテキストモードコンソールの場合には **Alt+F1** を使う必要があります。**F1** は、アップグレードが実行されている仮想ターミナルの番号と同じ番号のファンクションキーと置き換えてください。異なるテキストモードのターミナル間で切り替えを行うには、**Alt+左矢印** か **Alt+右矢印** も使えます。

## 特定のパッケージにする特別な注意

ほとんどの場合、**lenny** と **squeeze** 間でパッケージのアップグレードは順調に行われるはずですが、稀なケースとして、アップグレード前、あるいはアップグレード中に調整が必要な場合があります。以下にパッケージごとの詳細を記述します。

## Evolution

(GNOME デスクトップのメールクライアント) **Evolution** は、バージョン **2.22** から **2.30** へ更新されています。これにより、パッケージがローカルデータ用に利用している保存形式が更 されており、**evolution** が動作中にアップグレードが実施されると、データを損失する可能性があります。々 関連コンポーネントがバックグラウンドで動作しているため、アプリケーションの終了だけでは十分ではありません。々 問題の可能性を避けるため、**squeeze** へのアップグレード開始前に、デスクトップ環境を完全に終了しておくことをお勧めします。

アップグレード手順の一部として、**evolution** は関連プロセスが動作しているかをチェックし、終了することを勧めます。それから二回目のプロセスチェックが行われます。そしてもし必要であれば、残っているプロセスを終了するか、手動で状況を解決するためにアップグレードを途中で終了するかの選 が提示されます。

## カーネルと関連パッケージのアップグレード

このセクションでは、カーネルのアップグレード方法を明 し、このアップグレードに際して生じる可能性がある問題点を明確にします。**Debian** で提供されている **linux-image-\*** パッケージのいずれかをインストールしても、カスタマイズしたカーネルをソースからコンパイルしてもかまいません。

このセクションに書かれている多くの情報は、ユーザが **Debian** のモジュラーカーネルのいずれかを **initramfs-tools** や **udev** とともに使用しているのを前提としている、ということに注意してください。**initrd** を必要としないカスタムカーネルを使用するのを選 した場合や、**initrd** 生成ユーティリティとして異なるものを使用している場合は、このセクションの情報の一部は適切ではないかもしれません。

## カーネルメタパッケージのインストール

**lenny** から **squeeze** への **dist-upgrade** を実行する際には、新しい **linux-image-2.6-\*** メタパッケージをインストールすることを強くお勧めします。このパッケージは、**dist-upgrade** の過程で自動的にインストールされるかもしれません。次のように実行すると、このパッケージがインストールされたか確認できます。

```
# dpkg -l "linux-image*" | grep ^ii
```

何も出力されない場合は、新しい **linux-image** パッケージを手作業でインストールする必要があります。利用可能な **linux-image-2.6** メタパッケージの一覧を見るには次のように実行してください。

```
# apt-cache search linux-image-2.6- | grep -v transition
```

どのパッケージを選 すればよいのかわからない場合は、`uname -r` を実行し、似た名前をもつパッケージを探してください。例えば、コマンドの出力が '2.6.26-2-686' の場合は **linux-image-2.6-686** をインストールすることをお勧めします。利用可能なパッケージのうち最良のものを選ぶ手助けとして、次のように **apt-cache** を用いて各パッケージの **パッケージ明** の詳細版を参照してもよいでしょう。

```
# apt-cache show linux-image-2.6-686
```

インストールするカーネルイメージが決まったら、`apt-get install` でインストールします。新しいカーネルがインストールされたら、再起動できる時に再起動し、新しいバージョンのカーネルを有効にしてください。

少し勇気のある人には、Debian GNU/Linux 上で簡単に自分のカスタムカーネルをコンパイルするやり方があります。**kernel-package** ツールをインストールして、`/usr/share/doc/kernel-package` にあるドキュメントを読んでください。あるいは、**linux-source-2.6** で提供されるカーネルソースを使うこともできます。バイナリパッケージの構築には、ソース中の **Makefile** 中の **dep-pkg** ターゲットが使えます。この二つのやり方にはいくつか違いがあるので、それぞれのパッケージのドキュメントを確認してください。

可能なら、カーネルパッケージのアップグレードをメインの **dist-upgrade** と分けることで、一時的にでも起動不能なシステムにしてしまうことを極力避けられます。カーネルパッケージのアップグレードは、システムの最小アップグレード項で明 した最小アップグレードの手順の後以外では行うべきでないことに注意してください。

## デバイスの整列順序の更

**lenny** 以降では、ハードウェアの出 に関する新しいカーネルの仕組みは、システム上でデバイスが出 される順番を起動時ごとに えるかもしれず、与えられるデバイス名が影響を受けます。例えば、2つの異なるドライバと結び付いた 2つのネットワークアダプタがある場合、**eth0** と **eth1** が参照するデバイスは入れ替わるかもしれません。

ネットワークデバイスについては、通常 `udev` の `/etc/udev/rules.d/70-persistent-net.rules` で定義することでこの並べ替えを回避します。このルールは **lenny** で既に導入されているので、固定のネットワークデバイス名の恩恵を受けるために **squeeze** へのアップグレードの際に追加の作業をする必要はありません。しかし、この `udev` の仕組みは、割り当てられたネットワークデバイス名が特定のハードウェアに紐付けられているということに注意してください。例えば、もしあなたが **squeeze** を導入したシステムでイーサネットアダプタを交換した場合、新しいアダプタは既存の名前ではなく新しいインターフェイス名を使います。既存のデバイス名を新しいハードウェアで使う場合は、`/etc/udev/rules.d/70-persistent-net.rules` から関連するエントリを削除する必要があります。

ストレージデバイスについては、**initramfs-tools** を使い、ストレージデバイスのドライバモジュールを現在読み込まれているのと同じ順番で読み込むように設定することで、この並び替えを回避することができるでしょう。しかし、**IDE** から **PATA** サブシステムへのディスクドライバの移行項で記述されている **Linux** カーネルのストレージサブシステムへの他の 更内容 を考慮すると、これは大抵は労力に見合わないので、`/dev/disk/by-uuid/` ディレクトリ中の **UUID** のエイリアス や `/dev/mapper/` 中の **LVM** デバイス名などの長きに渡って安定していることが保証されているデバイス名を代わりに使うことが推 されています。

## 起動タイミングの問題

**initramfs-tools** で生成した **initrd** を使用してシステムを起動する場合、時として、`udev` によるデバイスファイルの作成が、起動スクリプトの動作に間に合わないことがあります。

通常の症状は、「ルートファイルシステムがマウントできず起動に失敗し、デバッグシェルに落ちる」というものです。しかし、後でチェックしても、必要なデバイスはすべて `/dev` に存在しています。この症状は、ルートファイルシステムが **USB** ディスクや **RAID** にある場合、特に **LILO** を使用している場合によく見られます。

この問題を回避するには、ブートパラメータに `rootdelay=9` を指定してください。タイムアウトの値 (秒で指定します) は調整する必要があるかもしれません。

# システムの起動が **Waiting for root file system** とともにハングしてしまう

`/dev/sda` となってしまった `/dev/hda` の復旧手順

一部のユーザからの報告によれば、アップグレードを実行すると、システム再起動後にカーネルがシステムのルートパーティションを見つけられなくなるようです。

このような場合、システムの起動は、以下のメッセージを最後にハングアップしてしまいます:

```
Waiting for root file system ...
```

そして数秒後に、`busybox` プロンプトがむき出しとなります。

この問題は、カーネルのアップグレードによって新しい世代の IDE ドライバが導入される際に発生する可能性があります。古いドライバでは IDE ディスクの命名規則が `hda`、`hdb`、`hdc`、`hdd` となっていました。これに して、新しいドライバは、同じディスクにそれぞれ `sda`、`sdb`、`sdc`、`sdd` という名前をつけます。

問題は、アップグレードによって、新しい命名規則を考慮に入れた、新しい `/boot/grub/menu.lst` ファイルが生成されないときに生じます。起動中に、カーネルが出せないシステムルートパーティションを、`Grub` がカーネルに渡してしまうのです。`/etc/fstab` がそれに応じて更新されなかった場合、ファイルシステムのマウント中にも発生します。しかし、`squeeze` へのアップグレード作業では、自動的に両方の状況がカバーされるはずですが。

アップグレード後にこの問題に遭遇した場合は、アップグレードの後で問題を解決するには項に飛んでください。アップグレード前にこの問題を防ぎたい場合は、このまま読み進めてください。

## アップグレードの前に問題を防ぐには

起動のたびに化しない識別子をルートファイルシステムに用いると、この問題を完全に防ぐことができます。識別子を用いる方法としては、2つが考えられます。ファイルシステムにラベルをつける方法と、ファイルシステムの汎用一意識別子 (UUID) を用いる方法です。これらの方法は、`Debian` では `etch` のリリースからサポートされています。

2つの方法のどちらにも長所と短所があります。ラベルをつける方法は可読性が高いのですが、マシン上の別のファイルシステムに同じラベルがついている場合に、問題が発生する可能性があります。見た目はよくありませんが、UUID を用いる方法では、2つの UUID が衝突することはまずありません。

以下の例では、ルートファイルシステムが `/dev/hda6` 上にあるものと仮定します。また、システムには動作する `udev` がインストールされており、ファイルシステムは `ext2` または `ext3` であると仮定しています。

ラベルをつける方法は、次のようにして実行します:

次のコマンドを実行して、ファイルシステムにラベルをつける (名前は 16 文字未満でなければなりません): **e2label /dev/hda6 rootfilesys**  
**/boot/grub/menu.lst** を編集して、次のような行を:

```
# kopt=root=/dev/hda6 ro
```

次のように更 する:

```
# kopt=root=LABEL=rootfilesys ro
```

### 注意

行頭の # を削除しないでください。この文字は行頭にある必要があります。

**update-grub** コマンドを実行して、**menu.lst** 内の **kernel** 行を更新する。  
**/etc/fstab** を編集し、/ パーティションをマウントするための、例えば次のような行を:

```
/dev/hda6 / ext3 defaults,errors=remount-ro 0 1
```

次のように更 する:

```
LABEL=rootfilesys / ext3 defaults,errors=remount-ro 0 1
```

ここで問題となる更 は最初のコラムです。この行の残りのコラムは更 する必要はありません。

UUID を用いる方法は、次のようにして実行します:

次のコマンドを発行して、ファイルシステムの汎用一意識別子を調べる: **ls -l /dev/disk/by-uuid | grep hda6** 。あるいは **blkid /dev/hda6** としても構いません。

**/dev/disk/by-uuid** ディレクトリの中身の一覧を表示すると、以下によく似た行が得られるでしょう:

```
lrwxrwxrwx 1 root root 24 2008-09-25 08:16 d0dfcc8a-417a-41e3-ad2e-9736317f2d8a -> ../../hda6
```

**blkid** を使った場合は、以下に似た出力が得られます:

```
/dev/hda6: UUID="d0dfcc8a-417a-41e3-ad2e-9736317f2d8a"  
TYPE="ext3"
```

UUID は、**/dev/hda6** を指しているシンボリックリンクの名前、すなわちこの場合は **d0dfcc8a-417a-41e3-ad2e-9736317f2d8a** です。

### 注意

あなたのファイルシステムの UUID は別の文字列でしょう。  
**/boot/grub/menu.lst** を編集して、次のような行を:

```
# kopt=root=/dev/hda6 ro
```

UUID を代わりに使います:

```
# kopt=root=UUID=d0dfcc8a-417a-41e3-ad2e-9736317f2d8 ro
```

### 注意

行頭の # を削除しないでください。この文字は行頭にある必要があります。

## アップグレードの後で問題を解決するには 解決方法 1

この方法が適用可能なのは、起動したいエントリを選ぶためのメニューインタフェースを **Grub** が表示してくれる場合のみです。そのようなメニューが現れない場合は、カーネルが起動する前に **Esc** キーを押して、メニューを表示させてみてください。このメニューに入れない場合は、解決方法 2項や解決方法 3項を試してみてください。

**Grub** のメニューにおいて、起動したいエントリをハイライトします。その上で **e** キーを押して、このエントリに関するオプションを編集します。オプションはおそらく次のようになっているでしょう:

```
root (hd0,0)
kernel /vmlinuz-2.6.32-5-686 root=/dev/hda6 ro
initrd /initrd.img-2.6.32-5-686
```

次の行をハイライトします:

```
kernel /vmlinuz-2.6.32-5-686 root=/dev/hda6 ro
```

**e** キーを押して、**hdX** を **sdX** に置換します (**X** は、システムによって異なりますが、**a**、**b**、**c**、**d** のいずれかです)。上の例では、置換後にこの行は次のようになります:

```
kernel /vmlinuz-2.6.32-5-686 root=/dev/sda6 ro
```

その上で、**Enter** を押して **更** を保存します。もし他の行にも **hdX** がある場合は、それらの行も同 **更** します。**root (hd0,0)** のようなエントリは **更** しないでください。すべての **更** が終わったら **b** キーを押してください。システムが通常どおり起動するはずですが。

システムが起動したら、この問題を恒久的に解決する必要があります。アップグレードの前に問題を防ぐには項に飛んで、提示されている 2つの手順のうちいずれかを実行してください。

## 解決方法 2

Debian GNU/Linux のインストールメディア (CD/DVD) から起動し、プロンプトが出たら **rescue** を選 して、レスキューモードを立ち上げます。言語と国、キーボードマップを選 して、その上で (成功するにせよ失敗するにせよ) ネットワークの設定を行います。暫くすると、ルートファイルシステムとして使用したいパーティションを選 ぶよう言われます。選 肢 として提示されるのは、次のようなものでしょう:

```
/dev/sda1
/dev/sda2
/dev/sda5
/dev/sda6
```

どのパーティションにルートファイルシステムが含まれているのか分かっている場合は、適切なものを選んでください。分からない場合は、とりあえず最初の選肢を選んでみてください。ルートファイルシステムのパーティションとして不適切だというエラーメッセージが出た場合は次の選肢を試し、それでもエラーメッセージが出た場合はさらに次の選肢を試し.....、というようにして探してください。順々にパーティションを試していても、それらのパーティションが壊れることはないはずです。ディスクに1つのオペレーティングシステムだけがインストールされている場合は、適切なルートファイルシステムパーティションが容易に見つかるはずです。ディスクに複数のオペレーティングシステムがインストールされている場合は、どのパーティションが適切なパーティションか、予めきちんと知っておいたほうがよいでしょう。

パーティションを選したら、々な選肢が提示されます。選したパーティションでシェルを実行するという選肢を選んでください。選んだときにエラーメッセージが出たら、他のパーティションで同じ操作を試してみてください。

これで、**/target** にマウントされたルートファイルシステムに、ユーザ **root** としてシェルでアクセスすることが可能になるはずです。ハードディスクの **/boot** ディレクトリや **/sbin** ディレクトリ、**/usr** ディレクトリの内容にもアクセスできる必要があります。これらは、それぞれ **/target/boot** や **/target/sbin**、**/target/usr** でアクセス可能になっているはずですが、必要であれば、これらのディレクトリを他のパーティションからマウントしてください (どのパーティションをマウントすればよいか分からない場合は、**/etc/fstab** を参照してください)。

アップグレードの前に問題を防ぐには項に飛んで、提示されている2つの手順のうちいずれかを実行し、この問題を恒久的に解決します。その上で、**exit** と入力してレスキュー用のシェルから け出し、**reboot** を選 してシステムを通常どおり再起動します (起動可能なメディアを取り出すのを忘れないでください)。

## 解決方法 3

お気に入りの LiveCD ディストリビューション (例えば Debian Live、Knoppix、Ubuntu Live など) から起動します。

`/boot` ディレクトリが置かれているパーティションをマウントします。どのパーティションに `/boot` ディレクトリがあるか分からない場合は、**`dmesg`** コマンドの出力を用いて、ディスクが `hda`、`hdb`、`hdc`、`hdd`、`sda`、`sdb`、`sdc`、`sdd` のいずれであるか調べてください。どのディスクを調べればよいか分かったら、次のコマンド (ディスクが `sdb` である場合の例) を発行してそのディスクのパーティションテーブルを表示し、適切なパーティションを見つけてください: **`fdisk -l /dev/sdb`**  
`/mnt/boot/grub/menu.lst` ファイルを編集します。ただし、適切なパーティションを `/mnt` にマウントしており、このパーティションが `/boot` ディレクトリやその内容を含んでいるとします。

次のようなセクションを見つけます:

```
## ## End Default Options ##

title          Debian GNU/Linux, kernel 2.6.32-5-686
root           (hd0,0)
kernel        /vmlinuz-2.6.32-5-686 root=/dev/hda6 ro
initrd        /initrd.img-2.6.32-5-686

title          Debian GNU/Linux, kernel 2.6.32-5-686 (single-
user mode)
root           (hd0,0)
kernel        /vmlinuz-2.6.32-5-686 root=/dev/hda6 ro single
initrd        /initrd.img-2.6.32-5-686

### END DEBIAN AUTOMAGIC KERNELS LIST
```

`hda`、`hdb`、`hdc`、`hdd` をそれぞれ、`sda`、`sdb`、`sdc`、`sdd` に置換します。次のような行は更 しないでください:

```
root           (hd0,0)
```

システムを再起動し、LiveCD を取り出します。これでシステムは適切に起動するはずですが。

システムが起動したら、アップグレードの前に問題を防ぐには項で提示されている 2 つの手順のうちいずれかを実行して、この問題を恒久的に解決します。

## 次期リリースへの準備

アップグレードの後で、次期リリースに向けてできるいくつかの準備があります。

時代 れ (Obsolete) のパッケージ項で明 するように、時代 れ (obsolete) のパッケージや、使用していないパッケージを削除してください。それらのパッケージが使用する設定ファイルを確認し、パッケージの完全削除 (**`purge`**) によって、設定ファイルも含めて削除することを討 してください。

## GRUB 2 へアップグレードする

アップグレード中、GRUB 2 を「チェーンロード」するオプションを進められることでしょう。これは GRUB Legacy をブートローダとして優先したままですが、GRUB 2 を読み込むオプションをそこに追加し、Debian GNU/Linux システムをそこから起動するようにします。これによって、GRUB 2 を使うという恒久的な更 を加える前に、GRUB 2 があなたのシステム上で動くかどうかを確認できるようになります。

GRUB 2 が動くことが確認できたら、これをちゃんと使うように切り換えをすべきです。チェーンロード設定は一時的な利用のみを意図しています。**upgrade-from-grub-legacy** を実行すれば、この切り換えが行えます。

GRUB マニュアルには、GRUB Legacy と GRUB 2 の違いについてより詳細な情報 [<http://www.gnu.org/software/grub/manual/grub.html#Changes-from-GRUB-Legacy>] が載っています。このうちの幾つかは、複雑な設定の場合には必要な変更かもしれません。ブートローダの設定をいじっていない場合は、これ以上何もすべきことはありません。

## 止予定のコンポーネント

Debian GNU/Linux の次期リリース 7.0 (コードネーム **wheezy**) では、いくつかの機能が止されます。システムを 7.0 へと更新する際の手間を省くため、ユーザは他の代替へ移行する必要があります。

以下の機能が止予定となっています:

**OpenVZ** と **Linux-Vserver**: Debian GNU/Linux 6.0 は、メインライン外の Linux カーネルの仮想化機能を含む最後のリリースになります。これは **OpenVZ** と **Linux-Vserver** の機能は非推奨 (**deprecated**) だとみなされ、ユーザが、**KVM** や **Linux** コンテナ、あるいは **Xen** のような **linux-2.6** の開発元で統合の仮想化ソリューションへ移行しなければならないことを意味します。

**gdm** パッケージ (**GNOME** ディスプレイマネージャ バージョン 2.20) は、書き直されたバージョンの **gdm3** によって時代遅れなもの (**Obsolete**) となっています。詳細については **GNOME** デスクトップに関する更新とサポート項を参照してください。

# 時代 れ (**Obsolete**) のパッケージ

**squeeze** では、数千個の新規パッケージが導入された一方で、**lenny** にあった 4,000 個以上の古いパッケージの破棄や削除が行われています。これら時代 れのパッケージをアップグレードする手段は提供されていません。時代 れのパッケージを使い続けても構いませんが、**Debian** プロジェクトでは通常、**squeeze** がリリースされてから 1 年後にそのようなパッケージへのセキュリティサポートを打ち切ります。またセキュリティサポート期間内であっても、時代 れのパッケージへの他のサポートは通常は提供しません。利用可能な代替品があるのであれば、代わりにそれを使うようにすることをお勧めします。

パッケージがディストリビューションから削除された理由は、数多くあります——もう上流で保守されていない、そのパッケージの保守作業に興味を抱く **Debian** 開発者がもういない、提供していた機能が別のソフトウェア (または新しいバージョン) に取って代わられた、バグのために **squeeze** にはもう適さないと見なされた、などです。最後の場合は、当該パッケージが「不安定版」ディストリビューション内にはまだ存在していることがあります。

更新が完了したシステム内のどのパッケージが「時代 れ」なのかを出すのは、パッケージ管理用フロントエンドが当該パッケージにその旨のマークをつけてくれるので簡単です。**aptitude** を使っているのなら、当該パッケージが「止された、またはローカルで作成されたパッケージ」欄に列記されているのに気づくでしょう。**dselect** も同じようなセクションを提供しますが、表示される一覧はわずかに異なっています。

さらに、**lenny** で手作業でパッケージをインストールするのに **aptitude** や **apt-get** を使っていたのなら、手作業でインストールされたパッケージの記 が取られています。依存関係のみによって引きずられてインストールされたパッケージにして、依存元パッケージが削除されてもう不要となった場合に、時代 れのマークをつけることができます。 **aptitude** と **apt** は、**deborphan** とは異なり、手作業でインストールしたパッケージには時代 れのマークをつけません (依存関係によって自動でインストールされたものにはマークをつけます)。自動的にインストールされたがもはや使われていないパッケージを削除するには、以下を実行してください:

```
# apt-get autoremove
```

**Debian** バグ追跡システム [<http://bugs.debian.org/>]は、パッケージが削除された理由についての追加情報を提供してくれることがよくあります。そのパッケージ自体と [ftp.debian.org](http://bugs.debian.org/cgi-bin/pkgreport.cgi?pkg=ftp.debian.org&archive=yes) 擬似パッケージ [<http://bugs.debian.org/cgi-bin/pkgreport.cgi?pkg=ftp.debian.org&archive=yes>]の両方の、アーカイブ化されたバグ報告を調べてください。

非推 パッケージ (**obsolete**) の一覧:

**plone** コンテンツマネジメントシステム。**Unified Installer for Linux** を使ってほしい、という開発者の要求に従って行われました。**Unified Installer for Linux** は、開発者らが唯一サポートしているとみなされるデプロイメントプラットフォームです。**Debian GNU/Linux** システム上で **Plone** をインストールするのにお勧めのツールは、<http://plone.org/> からダウンロードできる **Unified Installer** です。

**nessus** 脆弱性スキャンサーバと関連のライブラリおよびその他のソフト。

**openvas-server** と **openvas-client** を含む **OpenVAS** によって提供されるソフトウェアがあるため、非推奨 (**deprecated**) となりました。自動的にアップグレードする方法はないので、手動で **OpenVAS** をインストールして **Nessus** のサービス設定 (ユーザ、証明書など) を **OpenVAS** に移植する必要があります。

**postgresql-8.3** :後 となるパッケージは **postgresql-8.4**

**mysql-server-5.0** :後 となるパッケージは **mysql-server-5.1**

**python2.4** :後 となるパッケージは **python2.6**

**sun-java5-jre** や **sun-java5-bin** を含む Java 5 ソフトウェアは、Java 6 が後となります:**sun-java6-jre** とその関連パッケージです。

**apt-proxy** が、今後提供されなくなったので、このツールの代替には **apt-cacher-ng**、**apt-cacher**、**approx** があります。自動的にアップグレードする方法は存在していないので、**apt-proxy** のユーザは、手作業でこれらのパッケージのどれかをインストールすることによって代替品に移行が可能になります。

**squeeze** では、いくつかの **Xorg** のビデオドライバが時代 来 (**obsolete**) なため利用できなくなりました。これには以下が含まれます:**xserver-xorg-video-cyrix**、**xserver-xorg-video-i810**、**xserver-xorg-video-imstt**、**xserver-xorg-video-nsc**、**xserver-xorg-video-sunbw2**、**xserver-xorg-video-vga**。これらはアップグレード中に削除されるでしょう。ユーザは、代わりに **xserver-xorg-video-all** をインストールするべきです。

**lenny** で起動時にスプラッシュ画面を表示するのに使われていたユーティリティ **usplash** は利用できなくなっています。**plymouth** に置き換えられています。

## ダミーパッケージ

**lenny** の一部のパッケージは **squeeze** では複数のパッケージに分割されていますが、これは大半がシステムの保守性を改善するためです。このような場合のアップグレード手順を容易にするために、**squeeze** はしばしば「ダミーの」パッケージ——**lenny** での古いパッケージと同じ名前で、新規パッケージが自動的にインストールされるような依存関係を備えた空のパッケージ——を提供しています。これらの「ダミー」パッケージはアップグレード後は **Obsolete** 扱いとされ、安全に削除することができます。

(すべてではないのですが) 大半のダミーパッケージの明 には、その目的が記されています。しかしダミーパッケージの明文 は統一されていないので、自分のシステム内のダミーパッケージを出すには、**deborphan** を **--guess\*** オプションつきで使うのが便利でしょう (例: **--guess-dummy**)。一部のダミーパッケージは、アップグレード後に削除されることを意図したものではなく、プログラムのどのバージョンが現在利用可能な最新版かを長期にわたって追跡するのに使われる、ということに注意してください。

# 第5章 **squeeze** で知っておくべき問題点 生じうる問題

新しいリリースに導入された化には当然のように副作用がつきもので、どこか他の場所でバグを出してしまうこともあります。この章では、現時点で私たちが知っている問題点を記載しています。正誤表・関連パッケージの付属文書・バグ報告や、もっと読みたい項で触れられているその他の情報も読んでください。

## IDE から **PATA** サブシステムへのディスクドライバの移行

新しいバージョンの **Linux** カーネルは、いくつかの **PATA (IDE)** コントローラにして異なったドライバを提供します。ハードディスクや **CD-ROM**、テープデバイスの名前が変わるかもしれません。

これからは、設定ファイル中のディスクデバイスは、デバイス名よりもラベル (**label**) か **UUID (unique identifier)** で識別することをお勧めします。これは、古いカーネルバージョンと新しいカーネルバージョンの双方で動作します。**squeeze** バージョンの **Debian** のカーネルパッケージへのアップグレードに際して、**linux-base** パッケージは、**Debian** に含まれている色々なブートローダを含め、システム上のファイルシステム関連パッケージの大半が利用する設定ファイル中で、この換を実施してくれます。システム設定を自動的に更新しないようにした、あるいは **Debian** のカーネルパッケージを使っていない場合は、システムが確実に起動できるようにするため、次のシステム再起動の前にデバイスの **ID** を自分で更新しておく必要があります。

## **mdadm** メタデータ形式の更は最新の **Grub** を必要とします

以下は、**mdadm 3.x** のデフォルト値で作成した、あるいはメタデータバージョンが明示的に **-e** を使って設定された **RAID** デバイスから、**grub-pc** ブートローダを使って直接カーネルを読み込みたいユーザのみに当てはまります。つまり、これは **Debian squeeze** のインストール中、あるいはインストール後に作成されたすべてのデバイスに当てはまります。以前のバージョンの **mdadm** を使って作成されたアレイと、コマンドラインオプション **-e 0.9** で作成された **RAID** は影響を受けません。

バージョン **1.98+20100720-1** 以前の **grub-pc** は直接 **1.x** メタデータ形式の **RAID** から直接起動できなくなります (新しいデフォルトは **1.2** です)。ちゃんとシステムが起動できるようにするには、**Debian squeeze** で提供されている **grub-pc 1.98+20100720-1**、あるいはそれ以降のバージョンを必ず利用するようにしてください。起動できなくなったシステムは **Super Grub2 Disk** [<http://www.supergrubdisk.org/super-grub2-disk/>]、または **grml** [<http://grml.org/>] を使うと救出できるかもしれません。

## 新しい **libdb** による **pam\_userdb.so** の非互換性

**libdb3** で作成された **Berkeley DB** バージョン **7** のファイルは、より新しい **libdb** バージョンでは読み取りできないものがあります (**bug#521860** [<http://bugs.debian.org/521860>] を参照してください)。回避策として、**db4.8-util** パッケージの **db4.8\_load** でファイルを作り直すことが可能です。

## **/bin/sh** の切り換え (**diversion**) に関する問題の可能性

以前にローカルで **/bin/sh** を切り換え (**diversion**) していた、あるいは **/bin/sh** のシンボリックリンクが **/bin/bash** 以外のものを指すように更していたという場合、**dash** または **bash** パッケージのアップグレードの際に問題に遭遇するかもしれません。これには、他のパッケージ (例えば **mksh**) が **/bin/sh** を引きいで、デフォルトのシステムシェルになる **更** が含まれることに注意してください。

このような問題が出てきた場合、ローカルでの切り換え (diversion) を削除して、/bin/sh へのシンボリックリンクとマニュアルページが bash パッケージによって提供されているファイルを指すようにしてから、**dpkg-reconfigure --force dash** を実行してください。

```
dpkg-divert --remove /bin/sh
dpkg-divert --remove /usr/share/man/man1/sh.1.gz

ln -sf bash /bin/sh
ln -sf bash.1.gz /usr/share/man/man1/sh.1.gz
```

## リソースの競合に関連したカーネルのポリシー 更

Linux カーネルでの `acpi_enforce_resources parameter` のデフォルト値が、「strict」に更されました。これによって、レガシーなセンサードライバでセンサーハードウェアにアクセスができなくなるものがでできます。回避方法の一つは、カーネルのコマンドラインオプションに「`acpi_enforce_resources=lax`」を加えることです。

## LDAP サポート

LDAP ライブラリ中で使われている暗号化ライブラリの機能は、TLS や SSL を使っている LDAP サーバに接続する際に、LDAP を使っていて有効になっている特を更しようとするプログラムの接続障害を引き起こします。**sudo**、**su**、**schroot** のような **libnss-ldap** を使っているシステムや、**sudo-ldap** のような LDAP 索を実行する **suid** プログラムに問題が発生します。

**libnss-ldap** パッケージを、すべての LDAP 索に分割されたデーモン (**nslcd**) を使っている新しいライブラリである **libnss-ldapd** に置き換えることをお勧めします。**libpam-ldap** の代替品は **libpam-ldapd** です。

**libnss-ldapd** が、NSS キャッシュデーモン (**nscd**) をインストール前にあなたの環境に適しているかどうかを評価するのを推している点にご注意ください。**nscd** の代替としては **unscd** が考えられます。

更なる情報は、バグ#566351 [<http://bugs.debian.org/566351>] およびバグ#545414 [<http://bugs.debian.org/545414>] で得られます。

## sieve サービスが IANA 割り当てのポート番号に移動しました

ManageSieve への IANA のポート番号割り当ては 4190/tcp で、そして IANA レジストリ [<http://www.iana.org/assignments/port-numbers>] によると **timsieved** や多くのディストリビューションでの他の **managesieve** ソフトウェアによって使われていた古いポート番号 (2000/tcp) は Cisco の SCCP の利用に割り当てられていました。

Debian netbase パッケージのバージョン 4.38 から、`/etc/services` ファイルにて **sieve** サービスはポート番号が 2000 から 4190 へ移動しました。

数字のポート番号ではなく **sieve** サービス名を使っているソフトウェアはすべて、サービスが再起動あるいは再読み込みされる、あるいは `/etc/services` が更新された直後に、新しいポート番号へ切り替わります。

これは Cyrus IMAP に影響します。DoveCot のような **sieve** が利用できる他のソフトウェアにも影響を及ぼすかもしれません。

ダウン時間の問題を避けるには、**Debian** を使っているメールクラスタの管理者には **Cyrus** (そしておそらく **DoveCot** も) の設定の確認をお勧めします。そして、突然サーバあるいはクライアントでポート **2000/tcp** から **4190/tcp** にサービスが移動するのを避ける方策を講じましょう。

特記しておくべきこと:

**/etc/services** は、何も更 をしていなかった場合のみ、自動的に更新されます。それ以外の場合は、**dpkg** による更 についての確認が表示されます。

もし望む場合は、**/etc/services** を編集して **sieve** のポート番号を **2000** に することもできます (ですが、これはお勧めしません)。

**/etc/cyrus.conf** あるいはその他利用中のメール/ウェブメールクラスタの関連の設定ファイル (例: **sieve** ウェブフロントエンド上のもの) を編集して、前もって固定のポート番号を強制的に使うようにすることが可能です。

**cyrus master** が同時に両方のポート (**2000** と **4190**) を受け取るように設定して、全く問題を受けないようにすることができます。これによって、ポート **2000** からポート **4190** へのよりスムーズな移行も可能です。

## KDE デスクトップ

**squeeze** は、**Qt 4** をベースにした次世代の **KDE** をフルサポートする **Debian** の初リリースとなります。多くの公式 **KDE** アプリケーションはバージョン **4.4.5** で、**kdepim** のみバージョン **4.4.7** となっています。更点 については、**KDE** プロジェクトからのアナウンス [<http://www.kde.org/announcements/>]を読むことでさらに確認ができます。

## KDE 3 からのアップグレード

**Debian 6.0** では、**KDE 3** デスクトップ環境はサポートの象外 となりました。アップグレード時に自動的に新しい **4.4** シリーズに置き換わります。これは大きな更 なので、可能な限り円滑なアップグレード作業を確約するため、ユーザは事前の注意を受けておく必要があります。

### 重要項目

システム上に有効な **KDE 3** セッションがある間にアップグレードを行うのは推奨されていません。そうしなかった場合は、作業によって動作中のセッションが動かなくなり、データ損失が引き起こされる可能性があります。

アップグレード後のシステムへの初回ログイン時に、既存のユーザは **kaboom** と呼ばれる **Debian-KDE** ガイド付きの移行手続きが表示されます。これは、ユーザの個人データの移行作業と、お望みであれば以前の **KDE** 設定のバックアップ作業を手助けしてくれます。詳細については、**Kaboom** のホームページ [<http://pkg-kde.alioth.debian.org/kaboom.html>] を参照してください。

**KDE 3** のデスクトップ環境はもうサポートされませんが、ユーザが個々の **KDE 3** アプリケーションをインストールすることはまだ可能です。これは **KDE 3** のコアとなるライブラリとバイナリ (**kdelibs**) と **Qt 3** がまだ **Debian 6.0** で利用可能だからです。ですが、これらのアプリケーションは、新しい環境にうまく統合されていないことに注意しておいてください。さらに、**KDE 3** と **Qt 3** のどちらについても **Debian** の次期リリースではサポートされないため、これらを使っているのであれば、新しいプラットフォームへソフトウェアを移植することを強くお勧めします。

## 新しい KDE メタパッケージ

先に記したように、**Debian 6.0** では新たな **KDE** 関連のメタパッケージ群が導入されます:

通常のデスクトップ用途には、**kde-standard** パッケージのインストールを強くお勧めします。**kde-standard** は、デフォルトで KDE Plasma デスクトップ [<http://www.kde.org/workspaces/plasmadesktop/>] と、一般的に使われているアプリケーション群から選ばれたものを導入します。

最小限のデスクトップ環境が欲しい場合は、**kde-plasma-desktop** パッケージをインストールして、手動に必要なアプリケーションを選択してください。これで、Debian 5.0 での **kde-minimal** パッケージとおおよそ同等になります。

小型の機器用には、KDE Plasma ネットブック [<http://www.kde.org/workspaces/plasmanetbook/>] と呼ばれるもう一つの環境があります。これは **kde-plasma-netbook** パッケージでインストールが可能です。Plasma ネットブックと Plasma デスクトップは同一のシステム上に同居でき、システム設定 (以前 KControl だったものの代替) でデフォルトを指定できます。

公式の KDE アプリケーションをフルセットで欲しい場合には、**kde-full** パッケージをインストールするのが良いでしょう。デフォルトで KDE Plasma デスクトップがインストールされます。

## GNOME デスクトップに関する更 とサポート

GNOME デスクトップ環境は、**lenny** に含まれていたバージョンから **squeeze** に含まれているバージョンまでに多くの更 が加えられました。さらに詳しい情報は GNOME 2.30 のリリースノート [<http://library.gnome.org/misc/release-notes/2.30/>] にあります。固有の事柄は以下に げられています。

### GDM 2.20 と 2.30

GNOME ディスプレイマネージャー (GDM) は、**lenny** からのシステムアップグレードに際しては、バージョン 2.20 のままになっています。このバージョンは、**squeeze** のメンテ期間中は保守されますが、これが最後のリリースです。新しくインストールされるシステムには、代わりに **gdm3** パッケージで提供される GDM 2.30 が設定されます。双方のバージョン間での非互換性のため、このアップグレードは自動的にには行われませんが、**squeeze** へのアップグレード後には **gdm3** のインストールが推 されています。この作業はコンソール、あるいは一つだけオープンになっている GNOME セッションから行う必要があります。GDM 2.20 からの設定は引き ぎされません。しかし、標準的なデスクトップの場合は単に **gdm3** をインストール刷るだけで十分です。

### デバイスその他の管理者限

デバイスに する特別な限 が、現在システムへ物理的にログインしているユーザには自動的に与えられています: ビデオ・オーディオデバイス、ネットワークの切り換え、電源の管理、デバイスのマウントなどです。**cdrom**、**floppy**、**audio**、**video**、**plugdev**、**powerdev** グループの利用は、もはや有用ではありません。詳細な情報については、**consolekit** のドキュメントを参照してください。

**root** 限 が必要なほとんどのグラフィカルなプログラムは、**gksu** ではなく、**PolicyKit** [<http://www.freedesktop.org/wiki/Software/PolicyKit>] を使うようになっています。ユーザ管理者限 を与えるお勧めのやり方は、ユーザを **sudo** グループに追加する事です。

### network-manager と ifupdown 間での動作の関係

**network-manager** パッケージのアップグレードの際、**/etc/network/interfaces** で DHCP をオプション無しで使うようにしていたインターフェイスは、このファイルでは無効にされ、代わりに **NetworkManager** が管理するようになります。つまり、**ifup** と **ifdown** コマンドは動作しなくなります。このようなインターフェイスは、代わりに **NetworkManager** のフロントエンドを使って操作できます。**NetworkManager** のドキュメント [<http://live.gnome.org/NetworkManager/SystemSettings>] を参照してください。

逆に、`/etc/network/interfaces` でインターフェイスが色々なオプションを設定されている場合、`NetworkManager` では無視されます。これは、特に `Debian` のインストール中に利用されている無線接続に当てはまります ([bug #606268 \[http://bugs.debian.org/606268\]](http://bugs.debian.org/606268) を参照してください)。

## グラフィック関連ソフトウェアの更

`Debian 6.0` の `X` 環境には大量の更 が加えられています。この章では、もっとも重要でユーザから見える点を げます。

### 時代 れ (**Obsolete**) となった **Xorg** ドライバ

`cyrix`、`imstt`、`sunbw2`、`vga` の `Xorg` ビデオドライバはもう提供されません。ユーザは、代わりに `vesa` や `fbdev` などの汎用ドライバへ切り換える必要があります。

古い `via` ドライバは既にメンテナンスされなくなっており、`openchrome` ドライバに置き換えられました。このドライバは、アップグレード後に自動的に利用されるようになります。

今回のリリースには `nv` および `radeonhd` ドライバがまだ存在していますが、利用は推奨されません (`deprecated`)。ユーザは、それぞれ `nouveau` および `radeon` ドライバを代替として使うのを討 すべきでしょう。

`calcomp`、`citron`、`digitaledge`、`dmc`、`dynapro`、`elo2300`、`fpit`、`hyperpen`、`jamstudio`、`magellan`、`microtouch`、`mutouch`、`palmax`、`spaceorb`、`summa`、`tek4957`、`ur98` `X` 入力ドライバは提供中止となっており、今回のリリースには含まれていません。これらのデバイスのユーザは、 するカーネルドライバと `evdev` `X` ドライバへと移行した方がよいでしょう。多くのシリアルデバイスについては、`inputattach` ユーティリティを使えば `evdev` `X` ドライバで認識できる `Linux` 入力デバイスに接続が可能です。

### カーネルモードセッティング (**Kernel mode setting**)

カーネルドライバがネイティブモード設定をサポートするようになりました。それぞれ `Intel (i830 以降)`、`ATI/AMD (オリジナルの Radeon から Radeon HD 5xxx 「Evergreen」 シリーズまで)`、`NVIDIA` グラフィックスチップセットが象 です。

`intel` `X` ドライバでは、旧式のユーザ空間モード設定は止 されており、最近のカーネルが必要になります。カスタムカーネルのユーザは、設定に `CONFIG_DRM_I915_KMS=y` が含まれているのを確認する必要があります。

### 入力デバイスのホットプラグ

`Debian 6.0` の `Xorg` `X` サーバは、入力デバイス (マウス、キーボード、タブレットなど) のホットプラグのサポートが改善されています。古い `xserver-xorg-input-kbd` および `xserver-xorg-input-mouse` パッケージは、`xserver-xorg-input-evdev` によって置き換えられました。このパッケージは、`CONFIG_INPUT_EVDEV` オプションが有効になったカーネルを必要とします。さらに、このドライバが提供するキーコードの一部は、それまで同じキーが していたのとは異なるものがあります。`xmodmap` や `xbindkeys` などのプログラムのユーザは、新しいキーコードに設定を調整する必要があります。

### **X server** の 「停止方法」

伝統的に、`Ctrl+Alt+Backspace` の組み合わせで `X` サーバを停止できました。この組み合わせはデフォルトでは有効にはならなくなりましたが、`keyboard-configuration` パッケージの再設定 (システム全体に影響)、またはデスクトップ環境のキーボード設定アプリケーションを使うことで、再度有効にできます。

# 第6章 Debian GNU/Linux に関する さらなる情報 もっと読みたい

Beyond these release notes and the installation guide, further documentation on Debian GNU/Linux is available from the Debian Documentation Project (DDP), whose goal is to create high-quality documentation for Debian users and developers. Documentation, including the Debian Reference, Debian New Maintainers Guide, and Debian FAQ are available, and many more. For full details of the existing resources see the Debian Documentation website [<http://www.debian.org/doc/>] and the Debian Wiki website [<http://wiki.debian.org/>]

それぞれのパッケージの文書は `/usr/share/doc/` パッケージにインストールされています。ここには、著作権情報、Debian 固有の詳細、開発元の文書すべて、などが置かれています。

## 助けを求めるには

Debian ユーザ向けのヘルプ・アドバイス・サポートなどは、いろいろな場所から得られます。しかしこれらを頼りにするのは、その問題について徹底的に文書を調べてからにしましょう。このセクションでは新しく Debian ユーザになった人向けに、これらを簡単に紹介します。

### メーリングリスト

Debian ユーザが最も興味を引かれるであろうメーリングリストは `debian-user` (英語) リストおよび `debian-user-言語` (各国語) リストでしょう。これらのリストの詳細や講読のしかたについては、<http://lists.debian.org/> を見てください。利用にあたっては、あなたの疑問に対する答えが以前の投稿ですでに答えられていないかどうか、アーカイブをチェックしてください。また標準的なメーリングリストのエチケットに従うようにしてください。

## インターネットリレーチャット (IRC)

Debian には、Debian ユーザのサポートや援助のために専用の IRC チャンネルが OFTC IRC ネットワークにあります。このチャンネルにアクセスするには、お好みの IRC クライアントを [irc.debian.org](http://irc.debian.org) に接続し、`#debian` に `join` してください。

チャンネルのガイドラインに従い、他のユーザをきちんと尊重してください。ガイドラインは Debian Wiki [<http://wiki.debian.org/DebianIRC>] で参照できます。

OFTC についてさらに詳しく知りたい場合は、ウェブサイト [<http://www.oftc.net/>] を訪ねてみてください。

## バグを報告する

私たちは Debian GNU/Linux を高品質な OS にするよう努めていますが、だからといって私たちの提供するパッケージにバグが皆無というわけではありません。Debian の「オープンな開発体制」という考え方に合致し、また、ユーザにするサービスとして、私たちは報告されたバグに関するすべての情報を <http://bugs.debian.org/> にあるバグ追跡システム (Bug Tracking System: BTS) で提供しています。

もしディストリビューションや、その一部であるパッケージされたソフトウェアにバグを見つけたら、将来のリリースで修正できるよう、その問題点の報告をお願いします。バグを報告するには有効な電子メールアドレスが必要です。これをお願いしているのは、バグを追跡できるようにするため、そして追加情報が必要になった場合に開発者が報告者に連絡できるようにするためです。

バグ報告は、**reportbug** プログラムを使って送信することもできますし、電子メールを使って手で送ることもできます。バグ追跡システムに関する詳細やその使い方については、リファレンス文書 (**doc-debian** パッケージをインストールしてあれば `/usr/share/doc/debian` にあります) をお読み頂くか、またはバグ追跡システム [<http://bugs.debian.org/>]のウェブサイトからオンラインで入手することもできます。

## Debian に貢献する

Debian への貢献は専門家でなくてもできます。問題を抱えたユーザを、いろいろなサポート メーリングリスト [<http://lists.debian.org/>] で助けてあげることも、立派なコミュニティへの貢献です。開発 メーリングリスト [<http://lists.debian.org/>] に参加して、ディストリビューション開発に関する問題を見つける (そして解決する) ことも、もちろん非常に助けになります。Debian を高品質なディストリビューションに保つため、バグを報告して [<http://bugs.debian.org/>] その原因の特定や解決に際して開発者を助けてください。執筆が得意なら、文書 [<http://www.debian.org/doc/>] 作成や既存文書の自分の言語への翻訳 [<http://www.debian.org/international/>] に積極的に参加し、そこで貢献するのもよいでしょう。

もっと時間が自由になるなら、Debian に属するフリーソフトウェア集の一部を管理してみるのはいかがでしょうか。皆が Debian に入れてほしいと思っているソフトウェアを引き受けて管理するのは、特に価値の高い貢献です。これに関する詳細は、作業が望まれるパッケージのデータベース [<http://www.debian.org/devel/wnpp/>] をご覧になってください。Debian にはいくつかサブプロジェクトが存在しており、特定のアーキテクチャへの移植、Debian Jr. [<http://www.debian.org/devel/debian-jr/>]、Debian Med [<http://www.debian.org/devel/debian-med/>] などが進められています。これらのうち、あなたが興味を持っているグループに参加するのもよいでしょう。

いずれにしても、あなたが何らかの形でフリーソフトウェアコミュニティに関わっているのなら、それがユーザとしてであれ、プログラマ、ライター、翻訳者のいずれとしてであれ、すでにあなたはフリーソフトウェア運動を助けてくださっているのです。貢献することは報いのあることですし、 しいことです。新しい人々に出会う機会も えます。きっと暖かな、 しい気持ちになれるはずです。

# 付 A. アップグレードの前に **lenny** システムを調整する

この付には、**squeeze** へアップグレードする前に **lenny** パッケージを確実にインストールしたりアップグレードする方法についての情報が述べられています。特定の状況でのみ必要となるでしょう。

## **lenny** システムのアップグレード

基本的には、これまで行ってきた **lenny** のあらゆるアップグレードと違いはありません。唯一異なるのは、ソースリストのチェック項で明するように、パッケージリスト内に **lenny** への参照がまだ含まれているのを確認する必要があります。

**Debian** ミラーを使用してシステムをアップグレードする場合、システムは自動的に最新の **lenny** ポイントリリースへとアップグレードされます。

## ソースリストのチェック

`/etc/apt/sources.list` 内で `'stable'` を指定している行があるなら、効率よく **squeeze** を「使う」用意ができています。もしアップグレードの準備がまだできていない場合には、これはお望みの設定ではないかもしれません。すでに `apt-get update` を実行 済みでも、以下の手順に従えば問題なく元に戻すことができます。

**squeeze** からパッケージのインストールもしてしまっているなら、おそらくこれ以上 **lenny** からパッケージをインストールしても無意味でしょう。この場合、続けるかどうかを自分で決断しなければなりません。パッケージをダウングレードすることはできますが、その方法はここでは扱いません。

(`root` になってから) お気に入りのエディタで `/etc/apt/sources.list` を開き、`deb http:` や `deb ftp:` で始まるすべて行の中に「`stable`」が指定されているかどうかを調べてください。もしあるなら、`stable` を **lenny** に更してください。

`deb file:` で始まっている行があるなら、その行が指定している場所が **lenny** か **squeeze** のどちらのアーカイブなのかを独力で調べなければなりません。

### 重要項目

`deb cdrom:` で始まっている行は、に更しないでください。更するとその行は無効になって、もう一度 `apt-cdrom` を実行しなければならなくなるでしょう。`'cdrom'` ソースが「`unstable`」を指定していても心配しないでください。混乱するかもしれませんが、これで正常なのです。

更 げ たら、ファイルを保存してから

```
# apt-get update
```

と実行して、パッケージリストを更新してください。

## 古いロケールから **UTF-8** へアップグレードする

システムがローカライズされており、**UTF-8** ベースではないロケールを使っている場合、システムが **UTF-8** を使うように更 するのをよく考えるべきです。過去、**UTF-8** ではないロケールを使っている場合のみ明らかになるバグがありました。デスクトップでは、このようなレガシーなロケールはライブラリ内部での醜いハックによってサポートされているので、まだレガシーロケールを使ってるユーザに してきちんとしたサポートを提供できません。

**dpkg-reconfigure locales**を実行すれば、システムのロケールを設定できます。どのロケールをシステムのデフォルトとして使うかの質問を提示された時、確実に **UTF-8** ロケールを選んでください。さらに、ロケールを再度調 して、ユーザの環境設定中でレガシーなロケール定義がされていないことを確認する必要があります。

# 付 **B.** リリースノート の 貢献者たち

たくさんの方々が発行ノートを手伝ってくれました。以下の方々もそうですが、他にもいらっしゃると思います。

Adam Di Carlo, Andreas Barth, Andrei Popescu, Anne Bezemer, Bob Hilliard, Charles Plessy, Christian Perrier, Daniel Baumann, Eddy Petrișor, Emmanuel Kasper, Esko Arajärvi, Pop Frans [FAMILY Given], Giovanni Rapagnani, Gordon Farquharson, Javier Fernández-Sanguino Peña, Jens Seidel, Jonas Meurer, Josip Rodin, Justin B Rye, LaMont Jones, Luk Claes, Martin Michlmayr, Michael Biebl, Moritz Mühlenhoff, Noah Meyerhans, 小林 儀匡 [FAMILY Given], 青木 修 [FAMILY Given], Peter Green, Rob Bradford, Samuel Thibault, Simon Bienlein, Simon Paillard, Stefan Fritsch, Steve Langasek, Tobias Scherer, Vincent McIntyre, W. Martin Borgert

この文書はたくさんの言語に翻訳されています。翻訳者に大きな感謝を捧げます!

日本語への翻訳は以下の方が行いました。やまね ひでき [FAMILY Given]

# 目次

シンボル

パッケージ

mrtrix, 5

nipype, 5

odin, 5

psychopy, 5

## A

Abiword, 2

Apache, 2

## B

BIND, 2

Blu-ray, 2

## C

CD, 2

Cherokee, 2

Courier, 2

## D

Debian Live, 5

Debian Med, 5

Debian Science, 5

Dia, 2

DocBook XML, 2

DVD, 2

## E

Ekiga, 2

Evolution, 2

Exim, 2

## F

Firefox, 2

## G

GCC, 2

GIMP, 2

GNOME, 2

GNUCash, 2

GNUmeric, 2

## K

KDE, 2

KOffice, 2

## L

LDAP, 4, 2

LILO, 15

Linux Standards Base, 2

Live システム, 5

LXDE, 2

## M

Mozilla, 2  
MySQL, 2

## **N**

NeuroDebian, 5  
neuroimaging research, 5

## **O**

OpenOffice.org, 2  
OpenSSH, 2

## **P**

packages  
  ants, 5, 5  
  approx, 23  
  apt, 1, 1, 5, 5, 6, 6, 22  
  apt-cacher, 23  
  apt-cacher-ng, 23  
  apt-proxy, 23, 23  
  aptitude, 3, 10  
  autofs, 12  
  autofs5, 12  
  bash, 1, 2  
  console-setup, 3  
  consolekit, 4  
  dash, 1  
  db4.8-util, 1  
  dblatex, 2  
  debian-goodies, 9  
  dhcp3, 12  
  discover, 3  
  doc-debian, 2  
  docbook-xsl, 2  
  evolution, 14, 14  
  firmware-linux, 3  
  gdm, 21  
  gdm3, 21, 4, 4, 4  
  gksu, 4  
  grub, 11  
  grub-pc, 3, 1, 1, 1  
  icedove, 2  
  iceweasel, 2  
  initramfs-tools, 2, 14, 15, 15  
  insserv, 3  
  isc-dhcp, 12  
  kaboom, 3  
  kde-full, 4  
  kde-minimal, 4  
  kde-plasma-desktop, 4  
  kde-plasma-netbook, 4  
  kde-standard, 4, 4  
  kdelibs, 3  
  kdepim, 3  
  kernel-package, 15  
  keyboard-configuration, 5  
  libnss-ldap, 4, 2, 2  
  libnss-ldapd, 4, 4, 2, 2  
  libpam-ldap, 4, 2  
  libpam-ldapd, 4, 4, 2

- linux-base, 1
- linux-image-\*, 14
- linux-image-2.6-686, 15
- linux-source-2.6, 15
- live-build, 5
- localepurge, 10
- madwifi-source, 12
- mdadm, 1
- mksh, 1
- mrtrix, 5
- mysql-server-5.0, 23
- mysql-server-5.1, 23
- nessus, 23
- netbase, 2
- network-manager, 4
- nipype, 5
- nscd, 2, 2
- odin, 5
- openvas-client, 23
- openvas-server, 23
- plone, 23
- plymouth, 23
- popularity-contest, 9
- postgresql-8.3, 23
- postgresql-8.4, 23
- psychopy, 5
- python-setuptools, 12
- python2.4, 12, 23
- python2.6, 12, 23
- release-notes, 1
- splashy, 3
- sudo-ldap, 2
- sun-java5-bin, 23
- sun-java5-jre, 23
- sun-java6-jre, 23
- tinc, 3
- udev, 11, 11, 11, 11, 11, xx, xx, 11, 14, 15, 15
- unscd, 2
- upgrade-reports, 1
- usplash, 23
- xmlroff, 2
- xserver-xorg-input-evdev, 5
- xserver-xorg-input-kbd, 5
- xserver-xorg-input-mouse, 5
- xserver-xorg-video-all, 23
- xserver-xorg-video-cyrix, 23
- xserver-xorg-video-i810, 23
- xserver-xorg-video-imstt, 23
- xserver-xorg-video-nsc, 23
- xserver-xorg-video-sunbw2, 23
- xserver-xorg-video-vga, 23
- xsltproc, 2
- PHP, 2
- Pidgin, 2
- Postfix, 2
- PostgreSQL, 2

## **T**

- Thunderbird, 2

Tomcat, 2

**X**

Xfce, 2

# 用語集

ACPI	Advanced Configuration and Power Interface
ALSA	Advanced Linux Sound Architecture
APM	Advanced Power Management
CD	コンパクトディスク
CD-ROM	Compact Disc Read Only Memory
DHCP	Dynamic Host Configuration Protocol
DNS	ドメインネームシステム
DVD	デジタルバーサタイルディスク
GIMP	GNU Image Manipulation Program
GNU	GNU は Unix ではない (GNU's Not Unix)
GPG	GNU Privacy Guard
IDE	Integrated Drive Electronics
LDAP	Lightweight Directory Access Protocol
LILO	LIlinux LOader
LSB	Linux Standard Base
LVM	論理ボリュームマネージャ (Logical Volume Manager)
MTA	メール転送エージェント (Mail Transport Agent)
NFS	ネットワークファイルシステム
NIC	ネットワークインターフェースカード
NIS	ネットワークインフォメーションサービス
OSS	Open Sound System
RAID	Redundant Array of Independent Disks
RPC	リモートプロシージャコール
SATA	Serial Advanced Technology Attachment
SSL	セキュアソケットレイヤー (Secure Sockets Layer)
TLS	トランスポートレイヤーセキュリティ (Transport Layer Security)
USB	ユニバーサルシリアルバス (Universal Serial Bus)
UUID	汎用一意識別子 (Universally Unique Identifier)
VGA	Video Graphics Array
WPA	Wi-Fi Protected Access