



2005 年 東京エリア Debian 勉強会 , 関西出張版 事前資料

Debian 勉強会会場係 上川純一*

2005 年 10 月 29 日

* Debian Project Official Developer

目次

1	Introduction To Debian 勉強会	2
1.1	講師紹介	2
2	Debian Weekly News trivia quiz	3
2.1	2005 年 37 号	3
2.2	2005 年 38 号	4
2.3	2005 年 39 号	5
2.4	2005 年 40 号	6
2.5	2005 年 41 号	7
2.6	2005 年 42 号	8
2.7	2005 年 43 号	9
3	最近の Debian 関連のミーティング報告	11
3.1	東京エリア Debian 勉強会 9 回目報告	11
4	“claim” makes Debian better	12
4.1	本日の目的	12
4.2	Debian Bug Tracking System	12
4.3	BTS 用ツール	13
4.4	「重要度」と「タグ」	14
4.5	BTS の心得	15
5	Starting debconf translation	16
5.1	必要なもの	16
5.2	手順	16
6	debbugs internal	17
6.1	はじめに	17
6.2	データ形式	17
6.3	コード形式	20
6.4	そして何がおきたか	21
7	次回	24

1 Introduction To Debian 勉強会

上川純一



今月の Debian 勉強会へようこそ．これから Debian のあやしい世界に入るといふ方も，すでにどっぷりとつかっているといふ方も，月に一回 Debian について語りませんか？

目的として下記の二つを考えています．

- メールではよみとれない，もしくはよみとってられないような情報を情報共有する場をつくる
- まとまっていない Debian を利用する際の情報をまとめて，ある程度の塊として出してみる

また，東京には Linux の勉強会はたくさんありますので，Debian に限定した勉強会にします．Linux の基本的な利用方法などが知りたい方は，他でがんばってください．Debian の勉強会ということで究極的には参加者全員が Debian Package をがりがりを作りながらスーパーハッカーになれるような姿を妄想しています．

Debian をこれからどうするといふ能動的な展開への土台としての空間を提供し，情報の共有をしたい，というのが目的です．今回は違うこと言ってるかもしれませんが，御容赦を．

1.1 講師紹介

- やまね reportbug の使い手です．
- 上川純一 dpkg-dev-el, debian-el の使い手です．

2 Debian Weekly News trivia quiz

上川純一



ところで、Debian Weekly News (DWN) は読んでいますか？Debian 界隈でおきていることについて書いている Debian Weekly News. 毎回読んでいるといろいろと分かって来ますが、一人で読んでいても、解説が少ないので、意味がわからないところもあるかも知れません．みんなで DWN を読んでみましょう．

漫然と読むだけではおもしろくないので、DWN の記事から出題した以下の質問にこたえてみてください．後で内容は解説します．

2.1 2005 年 37 号

2005 年 9 月 13 日です．

問題 1. バグトラッキングシステムの見栄えで最近かわったのは何か

- A CSS を利用するようになった
- B DHTML になった
- C XHTML になった

問題 2. Debian UK で問題になったのは何か

- A メンバーが活動的でないこと
- B UK の経済状況がよろしくないこと
- C 商用利用をしようとした場合の Debian という名前の商標の利用の許可をする基準が不明確だったこと

問題 3. ソフトウェアを計測する，という論文で発表されたのは何か

- A Debian sarge には 2 億 3000 万行のソースコードが含まれている
- B Debian sarge の品質を計測した
- C Debian sarge の利用しやすさを計測した

問題 4. Joey Hess は testing に対して security 対応をすることを発表した．それに利用しているサーバはどれか

- A secure-testing.debian.net
- B security.debian.org
- C security.debuan.org

問題 5. /usr/doc をいまだにつかっているパッケージ数はどれくらいか

- A 100
- B 200
- C 500

問題 6. planet.debian.org をメーリングリスト経由で配布しようという意見に対して出た反論は

- A blog の内容は機密事項なので、メーリングリストで配布してほしくない
- B メーリングリストとして配布するとサーバの負荷が高くなる
- C blog の内容を永続的にメーリングリストのアーカイブとして保存されたくない

問題 7. `/usr/share/doc/パッケージ名/examples/` にあるファイルに実行権限をつけることについてはどうすべきか

- A サンプルは実行できるものは実行権限をつけるべき
- B サンプルなんてかざりなので実行しなくてよい
- C `/usr/share` 以下について実行権限をつけるのはこのましくなく、実行ファイルは `bin` におくべきだ。

問題 8. `sponsors.debian.net` が提供するサービスは何か

- A 金銭的寄付をつのるフィッシングサイト
- B 広告を配信し、広告収入を Debian プロジェクトの発展のために利用するサイト
- C まだメンテナになっていない人が管理しているパッケージについてスポンサーが必要な状況をトラッキングするシステム

問題 9. `1.0beta3` のようなベータ版のバージョン番号が `1.0` のような最終版のバージョン番号より低い、と `dpkg` が判定してしまう。この状況に関してメンテナはどう対応すべきか

- A 優先度の低いチルダ記号 `~` を利用して、`1.0~beta3` のような名前にする。ただまだアーカイブシステムが対応していないので、今後の改善が必要。
- B あきらめる
- C ベータ版はパッケージ化しない

問題 10. ソースのみのパッケージのアップロードを可能にするという提案についての反論は何か

- A バイナリが必要でなくなると、メンテナがテストをしなくなるのではないだろうか、という懸念がある
- B ソースのみだとパッケージインフラが破綻する
- C `katie` を改変するのが面倒

問題 11. BTS に任意のタグを追加できる機能が追加された、なんという機能が

- A `tagtag`
- B `たぐるんです`
- C `usertag`

2.2 2005 年 38 号

2005 年 9 月 20 日です。

問題 12. David Moreno Garza が `wnpp` にて `close` したバグレポートの数は

- A 729 のバグレポート
- B 100 のバグレポート
- C 123 のバグレポート

問題 13. International Conference on Open Source Systems に投稿された論文の中で説明されていた結果は

- A 開発者は短期間でどんどん入れ替わる
- B メンテナは実は幻想で、そんな人は存在しない
- C 長いあいだアクティブに活動し、パッケージの数も多くメンテナンスする

問題 14. Frank Lichtenheld が発表したのは、`non-free` なドキュメントを削除する処理を開始するということだっ

た．状況をトラッキングするために彼が利用したインフラは．

- A BTS の `usertags` 機能で `debian-release@lists.debian.org` ユーザのタグとして管理
- B Wiki ページ
- C CVS 管理のテキストファイル

問題 15. Software freedom day 05 で Debian-women が行って，結果として良かったので今後も継続することになったのは

- A `debian-women-new` IRC チャンネルがよい結果をもたらしたので，今後は `debian-women` チャンネルに新人を歓迎する時間帯というのをもうける
- B CD をたくさん焼いたら人気だった
- C Katie や BTSなどをインストールしてユーザがいじれるように提供したら人気だったので，今後もやる

問題 16. `init.d` スクリプトは現在直列に実行されているが，今後，並列実行を実装する際に便利だろうと思われる LSB 規格の仕様は

- A なんとなく並列に実行しても壊れないようにする仕様
- B 気持ちの中だけでは並列な年頃
- C `init` スクリプトの中で依存関係を記述できる仕様

問題 17. 新しいバージョンのパッケージにて問題が解決した場合の，バグレポートをクローズする方法でないのは何か

- A `changelog` でバグ番号を記述しアップロードする
- B バージョンヘッダを付けて，リクエストを `-done` アドレスに投げる
- C `btsclose` コマンドを利用する

問題 18. Marc Brockschmidt が説明した，新規メンテナプロセスの Front Desk の変更とは

- A 今後はより厳しい思想チェックを行う
- B Debian にコントリビュートしていることが要件になり，何もしていない場合は，応募が取り消される
- C 年齢制限を設けます

問題 19. `security.debian.org` で問題になったのは何か

- A セキュリティーアップデートが遅い
- B セキュリティーアップデートが嘘だった
- C `xfree86` のセキュリティアップデートがあまりにも高いネットワーク負荷を発生させてしまい，`security.debian.org` がサーバとして機能しなくなってしまった．

2.3 2005 年 39 号

2005 年 9 月 27 日です．

問題 20. Ben Hutching が `Debconf` について報告したのは

- A もう終わってしまった事は忘れる
- B 忘れ物がありました
- C DVD が入手可能になった

問題 21. `wiki.debian.org` への移行で特に手動の労力が必要だったのはどこか

A すでに wiki.debian.net から wiki.debian.org に移行してしまっているページがいくつかあったのでそれに対する手動の対処

- B kwiki から moinmoin ヘデータ形式の変更
- C ドメイン名の登録

問題 22. init の時点では/が read-only でマウントされているが、その時点でデータを保存するにはどうしたらよいか。

- A メモリファイルシステムを/run にマウントする
- B /mnt 以下にメモリファイルシステムをマウントする
- C /を rw にマウントしなおす

問題 23. グラフィックライブラリ GLU の実装が Debian 内で複数ある理由はなぜか

- A 一部のコードが一部のハードウェアでしか動かないという状況が続いているから
- B 複数のパッケージをメンテナンスしているほうがカッコいいから
- C メンテナの仲が悪いから

問題 24. Jeroen van Wolffelaar が提案したのは

- A libc5 を消す
- B libc6 を消す
- C libc6.1 を消す

問題 25. piuparts であきらかになる問題は

- A purge する際に、essential ではないパッケージに依存して、動作しないパッケージ
- B インストールしても動かないパッケージ
- C 使ってみて使いにくいパッケージ

2.4 2005 年 40 号

2005 年 10 月 4 日です。

問題 26. DPL チームが今後検討する予定の問題について記録する媒体として選択したのは

- A BTS
- B IRC bot
- C Wiki

問題 27. tetex 3.0 はどういう状況になっているか

- A 今後も入る見通しがない
- B うごかなくて困っている
- C experimental にアップロードされ、ライブラリのフリーズが完了したら unstable に入る

問題 28. Debian で配布する IA64 アーキテクチャ向けのカーネルについて Dann Frazier が SMP じゃないカーネルのサポートを削除しようとした、何故か

- A SMP じゃないといやだから
- B 時代は SMP です
- C IA64 で、SMP でないシステムがほとんどなく、あまりテストされていない

問題 29. Wolfgang Borgert によると planet.debian.org と、メーリングリストの利用方法の違いは

- A メーリングリストは古い技術なので今後はつかわなくなる
- B blog はフレームされないで意見を述べることでできるメディアだが、議論するのはメーリングリストで欲しい
- C planet.debian.org は安定していないので使わないで欲しい

問題 30. pbuttonsd は/dev/input/eventXX を利用しているが、どういう問題があったか

- A makedev が、最大 32 あるうちの 4 個しかデバイスファイルをつくっていなかったため、/dev を静的に管理しているユーザは一部の機能を利用できていなかった。
- B USB 接続ではうまく認識できなかった
- C 電源ボタンがおされたらアプリケーションがハングした。

2.5 2005 年 41 号

2005 年 10 月 11 日です。

問題 31. Debian security で改善したのは

- A バックエンドとフロントエンドのサーバを分割し、負荷に強い構成に変更した
- B 特定のユーザが負荷をかけられないようにスロットリングした
- C セキュリティーパッチをリリースしないことでサーバに負荷がかからないようにした

問題 32. Carlos Parra Camargo が報告したのは何か

- A Wiki が悪意をもったユーザにより書き換えられていたので前のバージョンを復活させた
- B Wiki がおもしろくないので改善しよう
- C Wiki サーバがダウンしている

問題 33. mozilla 1.7.8 に対してのセキュリティアップデートはどういう形でリリースされたか

- A 1.7.10 にバージョン 1.7.8 という名前をつけてリリースした
- B セキュリティーパッチをバックポートした
- C セキュリティーホールのある機能を全て disable にした

問題 34. 複数の chroot で同じユーザ情報を利用するのに利用できる方法でないのは

- A FUSE の shadow etc
- B LDAP
- C rm /etc/passwd

問題 35. ソースコードにローカルに適用したパッチをパッケージのアップグレード後も維持するためにはどうしたら一番楽か

- A 自分でがんばる
- B apt-src を利用する
- C パッチはあてない

問題 36. Jurij Smakov がリリースした文書は何か

- A Debian Users Handbook: Debian ユーザをどうあつかえばよいのか, が書いてある
- B Debian Developers Handbook: Debian Developer をどう扱えば良いか, が書いてある.
- C Debian Linux Kernel Handbook: Debian でカーネルがどうビルドされているのか, が書いてある

2.6 2005 年 42 号

問題 37. Elive って何?

- A 電子的に生きること
- B enlightenment ベースの LiveCD
- C Elise の新しいバージョン

問題 38. m68k について Steve Langasek が発表したのは

- A m68k を自分もつかいたい
- B m68k が Debian の移植版の中で一番素晴らしい
- C testing に入る条件として, m68k は無視することにした

問題 39. etch 向けの debian-installer の状況はどうか

- A もう全アーキテクチャについてインストールできることは確認した
- B もうすでに完全に動いている
- C まだ一部のアーキテクチャではビルドできない

問題 40. gnome1 のパッケージがビルドできなくなったのはなぜか

- A 古いから
- B libpng10 が削除されたから
- C gnome2 の時代がやっときたから

問題 41. Edd Dumbill が, sarge をインストールする際に, debian-installer を利用しているときにハードウェアの問題にあたったときに利用するように提案したのは

- A knoppix でハードウェア認識
- B ハードウェアを買い替える
- C Debian を使う事をあきらめる

問題 42. Oldenburg のミーティングの結果として, Debian security update で Branden Robinson が報告したのは

- A security.debian.org のバックエンドサーバが冗長構成になった
- B security.debian.org サーバが DNS のラウンドロビンで 3 台存在している構成がとれるようになった
- C security.debian.org がフィッシングサイトになった

問題 43. ソフトウェアに含まれている画像のライセンスに Creative Commons BY-SA ライセンスを利用したものは GPL のパッケージに含める事ができるか

- A やめたほうがよい
- B 可能
- C MJ Ray によると, 不可能なため, そのような場合は MIT ライセンスを利用したほうがよい

問題 44. Camm McGuire が libbfd にリンクするにはどうしたらよいのだ , と質問したときの Daniel Jacobwitz の回答は

- A libbfd は安定しているのでいくらでもリンクしてくれ
- B よくバイナリレベルの互換性は破壊されるので , binutils-dev にある libbfd.a をつかってくれ
- C 一般人は libbfd は使うな

2.7 2005 年 43 号

問題 45. Joerg Jaspert が NEW のパッケージを REJECT する理由で多いと指摘したのは

- A 読めないドキュメントが多い
- B おもしろくないパッケージが多い
- C debian/copyright が不正確なものが多い

問題 46. Steve Langasek が宣言した etch のリリーススケジュールによると , etch がリリースされるのは

- A 2005 年 12 月
- B 2006 年 6 月
- C 2006 年 12 月

問題 47. 今月 , Christian Perrier が宣言したかなり完成している , とコメントしていた debian-installer の機能は何か

- A sid をインストールするインストーラ
- B etch 向けのテキストモードのインストーラ
- C GTK を利用したグラフィカルインストーラ

問題 48. ypbind などが動的にポートを確保する場合 , その後に起動するサーバとポート番号がかぶる場合があるそれを回避する方法は

- A portreserve
- B 祈る
- C nis なんて使わない

問題 49. /etc/hosts に書いてある 127.0.0.1 のホスト名は現在の sid では何になるか

- A ホスト名
- B localhost.localdomain
- C localhost

問題 50. slang 用のモジュールパッケージ名は今後 'slang-モジュール名' という形になりそうだが , 従来はどういう名前だったか

- A sl モジュール名
- B モジュール名-slang
- C モジュール名

問題 51. pbuilder の開発体制にどのような変化があったか

- A 名前が変わりました
- B チームメンテナンス制をとるために、alioth に移動した
- C おもしろくなくなってきたのでもうやめます

問題 52. Daniel Ruoso が提案した Debian の移植版は

- A uclibc 移植版
- B minix 3.0 移植版
- C z80 移植版

問題 53. curl について openssl 版と gnutls 版の両方を提供するようになった、その理由は

- A GPL のプログラムが openssl とリンクしなくてすむように
- B 二種類あったほうが楽しいから
- C GNU のほうが openssl より凄いから

3 最近の Debian 関連のミーティング報告

上川純一



3.1 東京エリア Debian 勉強会 9 回目報告

前回開催した第 9 回目の勉強会の報告をします。
当日のタイムテーブルは下記でした

- 18:10- quiz
- 18:30- こたえあわせ
- 19:00- 休憩
- 19:10- たるさん
- 20:10- 上川
- 21:00- 宴会

10 月の第 9 回東京エリア debian 勉強会報告．今回は debbugs についての熱い話しを展開しました．今回の参加人数は登録者が 11 名くらいで，実際に参加したのが 9 名くらいでした．

DWN quiz に関しては，今回は小林さんが 1 問不正解で最高点数でした．

たるいしさんが apt-listbugs について説明しました．昨年の Asia Debian Mini Conf で発表した内容を説明して，実はそのころの TODO は進捗していない，ということを説明していました．osdn.debian.or.jp でミラーしているのですが，rsync もとはもともと master.debian.org だったが，同期できていないというバグ報告があって気づいて，merkel.debian.org に切替えたという話しがでていました．資料にあるグラフは 2003 年 9 月から 2004 年 10 月で今は一年たっているので，おそらく 7000 IP アドレスからの利用があるのだろう，と予想していました．また，中国で発表したときに，使ってみたら，バグの情報を取得するのに何分もかかり，ミラーサーバ必要だ，という話になった，がいまだになにもできていないとか，RSSViewer を使えば，自分のマシンに今入っているパッケージのセキュリティ関連のバグだけを見る，ということができる．実は便利かもしれない，とか．メンテナンスにあきてきたのでどうせならきなおしたいなぁという宣言もでていました．

上川が，Anthony Towns が Finland の debconf で発表していた内容と，その後にその発表に触発されて実装された Debbugs の新機能について話しました．おそらく Debbugs の仕様について日本語で記述した資料はこれが初めてなのではないでしょうか．新しい機能がいろいろと実装されており，apt-listbugs でも利用できそうな情報もあるので，たるいしさんの今後のハックに期待です．

4 “claim” makes Debian better

やまね



4.1 本日の目的

Debian BTS について理解を深める

- BTSって何さ？
- どういうときにするの？
- 何がいいの？
- 実際どうすればいいの？

そして立派なクレーマーとして認められる！

4.2 Debian Bug Tracking System

Debian 独自のバグ追跡システム。システムとしては debbugs という独自のものを利用

特徴

- Web から閲覧可能（まあ、最近のは皆そうですね）
- やり取りは基本的に全てオープン
- メールベースで作業が進む（ここは珍しいかも）
- かなり使い込まれてます。30 万件近くが登録済み。
redhat の bugzilla はこの半分ぐらいの件数

4.2.1 どんなときに BTS を利用しますか？

- パッケージングのバグに遭遇したとき
- アップグレードしたらよくわからない現象が起こるようになってしまったとき
- いつまで経っても security fix が提供されないとき
- 気の利いた機能を実装したのでパッチを取り込んでもらいたいとき
- 地味～な L10N な作業を取り込んでもらおうとき
- セキュリティホールの報告があったのでメンテナをせっつきたいとき

4.2.2 擬似パッケージ (pseudo package)

パッケージではないが、BTS で扱うためにパッケージとして扱うもの

- Web サイト
- wnpp – 作業が望まれるパッケージ (ITP も)
- インストールシステム などなど

<http://www.debian.org/Bugs/pseudo-packages.ja.html> 参照

Debian

お近くのサーバをお選び下さい:
アメリカ合衆国 Go

Debianについて ニュース Debianの入手 サポート 開発
サイトマップ 検索

Debian では、ユーザや開発者から寄せられたバグ報告の詳細を記録するバグ追跡システム (bug tracking system: BTS) を運用しています。バグにはすべて通し番号が与えられ、そのバグが処理されるまでその記録が保存されます。

バグ追跡システムに関する文書

- バグを報告する方法
- バグ報告のログを参照する方法
- 電子メールで既存のバグ報告を取り寄せる
- バグ追跡システムの使用方法に関する開発者情報
- 電子メールでバグ報告を操作することに関する開発者情報
- メールサーバリファレンスカード

WWW 上でバグ報告を閲覧する

番号でバグを検索する:

検索 ☐ mbox 形式で

以下の条件でバグを検索する: ☒ パッケージ名 ☐ ソースパッケージ ☐ パッケージメンテナの電子メールアドレス ☐ バグ報告者の電子メールアドレス ☐ 重要度 (severity) ☐ タグ

上記の項目が であるようなバグを検索する

検索

図 1 BTS のページ (<http://www.debian.org/Bugs/>)

ここでのポイント

あらゆる苦情・提案は BTS に集まる。誰も聞いていないところで文句を言うのではなく BTS すべし。

4.3 BTS 用ツール

4.3.1 reportbug/querybts

reportbug コマンド

簡単にバグレポート・レポートの検索が可能

- 対話的な操作が可能です。
- レポートはメールで飛びます。ポーンと。
- レポートは gnupg で署名も可能。まるでちゃんとした報告みたいに見えます。

querybts コマンド

バグレポートの検索に特化しています。

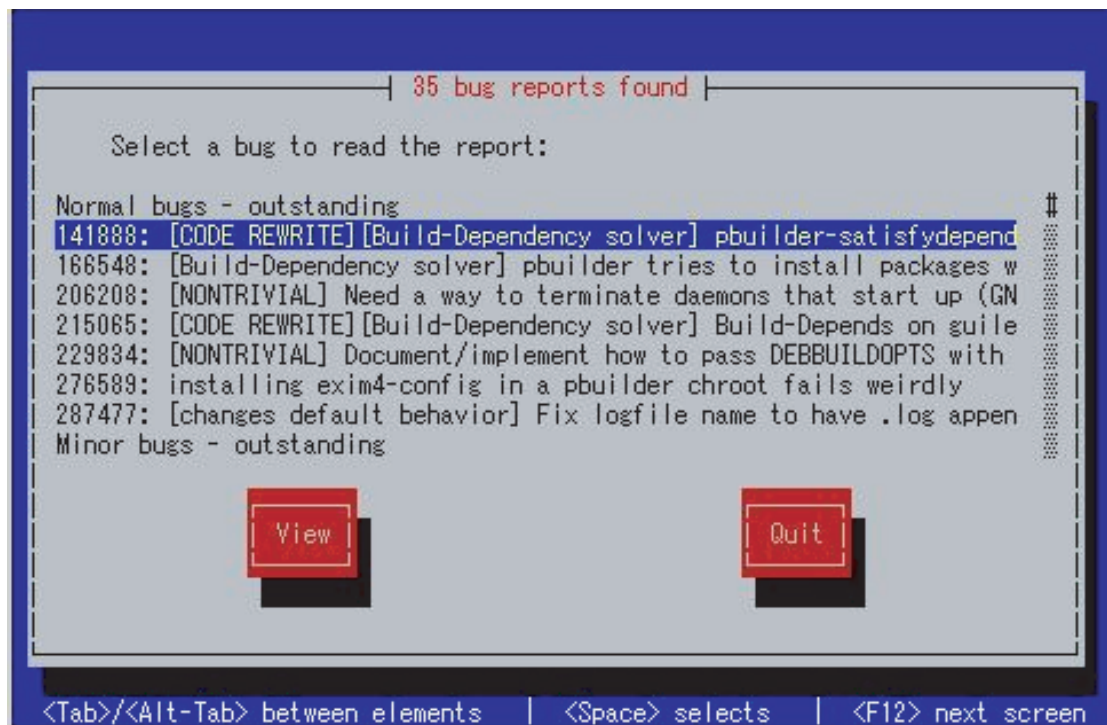


図 2 reportbug の画面

4.3.2 debugs-el

emacs ユーザは、debugs-el パッケージに含まれる debian-bug コマンドを使うこともできます。M-x debian-bug と入力すると、reportbug とよく似たやり方で、全ての必要事項を尋ねられます...らしい。(emacs 使ってないので不明)

4.4 「重要度」と「タグ」

4.4.1 Severity (重要度) レベル

<http://www.debian.org/Bugs/Developer#severities> 参照

- critical(致命的)
システム上の関係のないソフトウェア (またはシステム全体) を破壊する、重大なデータの欠落を引き起こす、または、そのパッケージをインストールしたシステム上でセキュリティホールが生じる場合。
- grave(重大)
問題のあるパッケージが使用できない、またはほとんど使用できない。またはデータの欠落を引き起こす、そのパッケージを使用するユーザのアカウントにアクセスを許してしまうセキュリティホールが生じる場合。
- serious(深刻)
Debian ポリシーに対して見すごせない違反がある (大まかに言うと、"must" や "required" の要件に違反している)、またはパッケージメンテナの意見としてそのパッケージがリリースに適していないと判断された場合。
- important(重要)
バグがパッケージの利用に大きく影響しており、対処しなければ誰にもまったく使用できない場合。
- normal(通常)
デフォルト値。通常のバグ。
- minor(軽度)

問題がパッケージの利用に影響しない、かつ修正はたいした事がないと思われる場合。

- wishlist(要望)

将来的な要望、主に設計上の理由により修正が非常に困難なバグ。

4.4.2 タグ

<http://www.debian.org/Bugs/Developer#tags> 参照

- patch(パッチ)

バグ報告に、バグを修正するためのパッチや簡単な手順が含まれています。パッチがあってもバグを適切に解決できない場合や別の問題を生じる場合は、このタグは使うべきではありません。

- security(セキュリティ)

このバグはパッケージのセキュリティ問題を説明します。ほとんどのセキュリティバグは、critical (致命的) や grave (重大) の severity (重要度) も設定すべきです。

- upstream(上流)

このバグは、パッケージの上流の部分に影響します。

- d-i(インストーラ)

このバグは、debian-installer に関するものです。インストーラの開発に関係するけれども、インストーラの直接の構成要素ではないパッケージに対するバグの場合、このタグを使ってください。

- L10n

このバグは、パッケージの地域化に関するものです。

- woody / sarge / sid / experimental

このバグは特に各ディストリビューションに加えられるものです。

4.5 BTS の心得

- 何よりも相手を尊重しよう

- 事実を端的に述べよう

- 環境の記述は多すぎず少なすぎずを目指そう

- バージョンやアーキテクチャぐらい書こう (面倒な人はツールを使いましょう)

- broken English でもいいや、と開き直ろう

- でも多少は体裁は整えておこう

- gnupg 使ってみるとか

- 定型シグネチャ使っておくとか

5 Starting debconf translation

やまね



<http://kmuto.jp/debian/po-trans/> に翻訳の状況が掲載されている。

5.1 必要なもの

- debconf-updatepo
po が最新のものをチェック
- msgfmt
文法があるかを確認

5.2 手順

1. po-debconf をインストール

```
apt-get install po-debconf
```

2. 翻訳したいパッケージのソースをインストール

```
apt-get source <packagename>
```

3. template.po をコピーして翻訳する

```
cd <packagename>/debian/po  
cp template.pot ja.po
```

4. 査読してもらう

debian-doc@debian.or.jp とかに投げる

5. BTS する

```
reportbug -A 翻訳したファイル -g  
ファイルを添付して GPG Sign してバグレポート送信
```

6 debbugs internal

上川



6.1 はじめに

この文書は Anthony Towns がフィンランドの debconf 5 で発表した内容を日本にて展開するための資料です。Anthony Towns の作成した英語の資料を省略して抜粋しています。また、それ以降に変更した事項について追記しています。

Debian Bug Tracking System (BTS) は、ほぼ Debian に特化したバグ報告の管理のためのシステムです。他のプロジェクトでも利用されていることもありますが、Debian でバグがパッケージベースで厳格に分類できることなどの特性が反映されているため、Debian プロジェクトのワークフローで使いやすいように作られています。^{*1}

規模としては、55000 以上の現在アクティブなバグ報告、231000 のアーカイブされたバグ報告を現在保持しています。毎週 1000 以上の新規のバグ報告が追加されています。ウェブインタフェースは追加された報告をすぐに反映しており、過去、ダウンタイムもほとんど発生していません。

Anthony Towns によると下記がバグトラッキングシステムの要件です。

- インタフェース：開発者がメールで操作できるようになっており、誰でもウェブで閲覧できるようになっている。
- パッケージベース：バグ報告をパッケージ別に高速に管理する必要がある
- スケーラビリティ：大量のバグ報告に対応できる必要がある
- 即時性：現在のバグの状態をすぐに報告してくれる必要があり、バグの状態が変更されたらすぐに反映される必要がある
- 安定性：継続して動作する必要がある。新規の機能がどんどん追加されたとしても。
- 公開：議論の内容に Debian コミュニティ全体として参加できるように、永続的な公開記録として保存される必要がある。

6.2 データ形式

バグデータベースのスプールの形式は下記です。リレーショナルデータベースなどは利用していません、スプールディレクトリ以下にほとんどのデータが格納されています。

各バグについて、ファイルはそれぞれ 4 個あります。サマリーファイルはメタデータを保存します。ログファイルは、そのバグに対して流れたメールを全て保存します。

status ファイルは互換性のためだけに存在しています。report ファイルは、最初のバグ報告のメールで、バグが close されるときに送信されるものです。

- /org/bugs.debian.org/spool
 - incoming/
 - * T.*
 - * S[BMQFUDU RC] *.*

^{*1} Debian のインフラと統合されており、changelog にバグ番号を記述してパッケージをアップロードしたらバグが修正されたと記録されるようになっていたりします。

```

    * R[BMQFDU RC] *.*
    * I[BMQFDU RC] *.*
    * G[BMQFDU RC] *.*
    * P[BMQFDU RC] *.*
- db-h/
    * 00/
        . ..
        . 314200.log
        . 314200.report
        . 314200.status
        . 314200.summary
    * ..
    * 99/
- archive/
    * 00/
        * ..
        * 99/
- index.db - index.db.realtime へのシンボリックリンク
- index.archive - index.archive.realtime へのシンボリックリンク
- nextnumber

```

6.2.1 incoming

incoming に来たメールは処理中、名前を変えます。

- T receive によってうけとられた
- S SPAM 確認待ち
- R SPAM 確認中
- I SPAM チェック通った
- G service か process スクリプトを通った
- P process 中

また、ファイル名の二つ目の文字はどのメールアドレスにメールが送信されてきたものなのかということを示します。ファイル名ののこりは、バグ番号と、一意な ID です。一意な ID を決定するのに現在は時間とプロセス番号を利用しています。

- B: 通常のバグ報告 submit@ 1234@
- M: -maintonly メーリングリストに投げない
- Q: BTS に登録しない -quiet
- F: アップストリームにフォワード -forwarded
- D: バグ終了 -done
- U: サブミッターにメール -submitter
- R: ユーザのリクエスト用インタフェース request@
- C: デベロッパーの制御用インタフェース control@

6.2.2 Status と Summary

status ファイルの中身は行ベースです．無い行については空行とみなします．このファイルは今後なくしていこうとしています．

- バグ報告者のメールアドレス
- 時間 (秒)
- サブジェクト
- 元のメールのメッセージ ID
- バグがアサインされているパッケージ
- タグ
- close した人のメールアドレス
- 上流のメールアドレスか URL(forward されたばあい)
- マージされているバグ番号
- severity

summary ファイルは RFC822 形式で，拡張可能になっています．現在 Format-Version: 2 と 3 の二つの形式があります．3 は，ヘッダについては RFC1522(MIME) のデコードされた形式になっています．

- Format-Version: このファイル形式のバージョン
- Submitter: バグ報告者のメールアドレス
- Date: 時間 (秒)
- Subject: サブジェクト
- Message-ID: 元のメールのメッセージ ID
- Package: バグがアサインされているパッケージ
- Tags: タグ
- Done: close した人のメールアドレス
- Forwarded-To: 上流のメールアドレスか URL(forward されたばあい)
- Merged-With: マージされているバグ番号
- Severity: severity
- Owner: バグの所有者

6.2.3 log ファイル

あらゆるメールが log ファイルには追記されていきます．また，メタデータも追記されていきます．残念ながら，メタデータは生の HTML で書かれており，またバージョンによって記述の仕方が変わっており，さらに悪いことに，古いバグの中にあるテキストは更新されていないため，機械的に処理することは難しくなっています．

また，コントロール情報は，行頭のエスケープコードにより切り替わります．メールの中にエスケープコードのような文字列が出て来たら，それは文字コード 030(8 進数) の文字を追加してエスケープします．

詳細は Debbugs::Log を見てください．

- kill-init: まだ一行も処理していません
- incoming-recv: 07: あとに go がくる，Received:行
- autocheck: 01: X-Debian-Bugs-...: までの無視されている行，autowait が次に来る
- html: 06: 生で表示すべき HTML
- recips: 02: メールの受取人，04 で分割されている
- go: 05: メールの文書

- go-nox: X: メールの文書, X ではじまる行
- kill-end: 03: メッセージの終り.
- autowait: go-nox があとにくる, 空行まで無視されるその他の情報.

6.2.4 Index ファイル

index ファイルは, pkgreport.cgi がどのパッケージにどのバグがわりあてられているかを確認するための情報です. 以前は, by-package.idx と by-severity.idx というのがあり, 高速化に貢献するはずだったのですが, 一年以上長い間生成されていなかったうえに, 生成されていなかったことに誰も気づかなかったので必要ないんじゃないだろうか, ということです.

データ形式としては下記のようになります. パッケージ, バグ番号, 時間, ステータス, メールアドレス, severity の順に書いた行が全てのバグに対して作成されています.

```
pbuilder 317998 1121196782 open [Junichi Uekawa <dancer@netfort.gr.jp>] normal
```

6.3 コード形式

debbugs は特に設計もされずに長い間パッチを累積してきました. ただ, 明確にわかれている部分はあって, メールを処理するコアのインタフェースのスクリプトと, ウェブを表示するための CGI 部分とで分離できます.

設定ファイルは全て/etc/debbugs にあります.

6.3.1 コアのスクリプト

メールを処理する部分があります.

- errorlib: ライブラリ
- receive: MTA からメールを受信する
- spamscan: 受信メールを SPAM チェックする
- processall: process と service にメールを分配する
- process: バグメールを処理する
- service: control@ と report@ メールを処理
- expire: close されてから 28 日過ぎたバグをエキスパイア処理する
- rebuild: index ファイルをリビルド

receive と rebuild 以外は cron から起動しています. 15 分に一回しか動作しません.

6.3.2 CGI スクリプト

CGI 関連は, errorlib 関数を活用している部分もありますが, ほぼ独立しています.

- bugreport.cgi: バグレポートを一つ表示
- pkgreport.cgi: パッケージやサブミッタなどでサマリを作成する
- pkgindex.cgi: パッケージや severity に対して数を表示
- common.pl: ライブラリとして利用

pkgreport.cgi はユーザが直接ウェブでたたくため, 特に速度が重要視される部分なので, 触る場合には注意してください.

6.3.3 ハックするには

debbugs のソースは CVS にあります. また, Debian Developer であれば, ミラーが merkel.debian.org の /org/bugs.debian.org 以下にあります.

6.4 そして何がおきたか

Anthony Towns の発表でどういう結果がもたらされたか見てみましょう。

6.4.1 バージョントラッキング

バグがどのバージョンで発見され、どのバージョンで修正されたのかというのをトラッキングできるようになりました。従来は発見されたバージョンだけが Version ヘッダで分かるようになったのですが、それ以外の情報も保持するようになりました。

<http://lists.debian.org/debian-devel-announce/2005/07/msg00010.html>

バグ番号を保持して操作するための BTS のコマンドは下記です。

```
close バグ番号 バージョン
reassign バグ番号 バッケージ バージョン
found バグ番号 バージョン
```

また、katie が変更され、バグを close するメッセージには、下記のヘッダが付くようになりました。それを BTS が処理して close されたバージョンを把握できるようになりました。

```
Source-Version: バグ番号
```

CGI に対して、version を指定すると、そのバージョンでの状態がでできます。329344^{*2}は、0.4 では open だったが、0.5 では close になったというのが <http://bugs.debian.org/cgi-bin/pkgreport.cgi?pkg=cowdancer&version=0.4> と <http://bugs.debian.org/cgi-bin/pkgreport.cgi?pkg=cowdancer&version=0.5> 二つのページを比較するとわかります。

</org/bugs.debian.org/spool/db-h/44/329344.summary> を見ると、メタデータとして保存されているのがわかります。

```
Format-Version: 2
Found-In: cowdancer/0.4
Done: Junichi Uekawa <dancer@debian.org>
Subject: cowdancer: cow-shell does not start, gives error
Date: 1127295198
Submitter: Francesco Potorti' <Potorti@isti.cnr.it>
Fixed-In: cowdancer/0.5
Package: cowdancer
Message-Id: <E1EIOYD-00031E-00@pot.isti.cnr.it>
Severity: grave
```

6.4.2 ユーザタグ

<http://lists.debian.org/debian-devel-announce/2005/09/msg00002.html>

request@bugs.debian.org に対して下記のようなメールをおくればタグが追加できます。

```
user aj@azure.humbug.org.au
usertag 18733 + good-reasons-to-run-for-dpl
usertag 18733 + still-cant-believe-it-finally-got-fixed
usertag 62529 + your-days-are-numbered
```

見る際には、users=でユーザを指定するとタグが見えるようになります。

<http://bugs.debian.org/cgi-bin/pkgreport.cgi?pkg=dlsip;users=dancer@debian.org>

また、tag でタグを指定して、users=でユーザを指定するとそのユーザで作成したタグを全て検索することができます。

<http://bugs.debian.org/cgi-bin/pkgreport.cgi?tag=ignore-for-now;users=dancer@debian.org>

6.4.3 バグ購読

バグ番号に対してメーリングリストのようにして利用できるようになりました。<http://lists.debian.org/debian-devel-announce/2005/07/msg00014.html>

^{*2} <http://bugs.debian.org/329344>

バグ番号-subscribe@bugs.debian.org にメールを出すと登録するようにメールがかえって来るので、それに返信すると、バグ番号に登録されます。

6.4.4 バグブロッカー

どのバグがどのバグによって邪魔されているのかというのをトラッキングするための機能が追加されました。

```
block 保留中のバグ番号 by 原因のバグ番号
unblock 保留中のバグ番号 by 原因のバグ番号
```

6.4.5 mindays maxdays

mindays, maxdays オプションが追加されました。バグ報告の報告されてからの日数で表示させるかさせないかを選択できるオプションです。

<http://bugs.debian.org/cgi-bin/pkgreport.cgi?maint=dancer@debian.org&maxdays=90> や <http://bugs.debian.org/cgi-bin/pkgreport.cgi?maint=dancer@debian.org&mindays=90> として入力できます。

6.4.6 バグ検索システム

Google が master.debian.org にとって DoS になるような検索の仕方をしていたので、現在 BTS は google の検索対象にははいていないので検索サービスが必要だろう、という話題が出ていました。

鵜飼さんが全文検索エンジンサービス (FABRE) を実装しましたが、まだ本格的に使われるような状態にはまだいたっていないようです。結構このサービスは負荷が高いのが問題になると思われます。<http://fabre.debian.net/>

6.4.7 debian-bugs.el はまだ動くのか

reportbug は vi ユーザを中心としたインタフェースになっていますが、Emacs を利用している人でも、debian-bugs.el を利用して debian BTS を操作することができます。

たとえば、debian-changelog-mode を使っている場合であれば、changelog のエントリーを自動生成するようになっています。メニューから Bugs close を選択するか、debian-changelog-close-bug でバグ番号を選択する (タブ補完がききます) と、下記のようなエントリーが作成できます。

```
dsh (0.25.6-2) unstable; urgency=low

* Bug fix: "How to control dsh timeout time?", thanks to Junichi Uekawa
(Closes: #281012).
* Bug fix: "allow exclusion of host from list of hosts.", thanks to
Junichi Uekawa (Closes: #289766).
* Bug fix: "dsh: -c -i hangs if no input under current design", thanks
to Charles Fry (Closes: #241531).
```

この機能は 207852^{*3} で上川の出したパッチが発端で実装されましたが、気づいたら正規表現が巨大になっています。

ひさしぶりにソースコードをみたら、現在の実装は、正規表現をつかいて HTML を解析しているため、何かイレギュラーなことがあった場合には、動作しなくなります。該当する関数は emacs-goodies-el:/elisp/debian-el/debian-bug.el(debian-bug-build-bug-menu) です。

BTS のフォーマットが変わったので、何かうごかなくなっていないかと心配していましたが、特にうごかないということはないようです。

みると submitter のメールアドレスがうまく解析できなかった場合には 'thanks to XXXX (closes: XXXX)' が追加されないという仕様になっています。たまにこれが発生するので、再現する条件をさがしてバグを直したいですね。

^{*3} <http://bugs.debian.org/207852>

```

(with-temp-buffer
 (message "Fetching bug list...")
 (call-process "wget" nil '(t t) nil "--quiet" "-0" "-"
  (concat
   "http://bugs.debian.org/cgi-bin/pkgreport.cgi?src="
   package))
 (message "Fetching bug list...done")
 (goto-char (point-min))
 (while
  (re-search-forward
   "\\(<H2.*</a>\\(\\(.+\\)</H2>\\)\\|\\(<li><a
href=\\\"\\(bugreport.cgi\\?bug=\\([0-9]+\\)\\)\\\">\\(#[0-9]+: \\(\\(.+\\)\\)</a>\\)\"
   nil t)
  (let ((type (match-string 2))
        ;;(URL (match-string 4))
        (bugnumber (match-string 5))
        (description (match-string 6))
        (shortdescription (match-string 7)))
    (cond
     (type
      (setq bugs-are-open-flag (not (string-match "resolved" type)))
      (save-excursion
       (set-buffer debian-bug-tmp-buffer)
       (insert "\\-\\n\\n" type "\\n\\n")))
     (t
      (setq bug-alist (cons (list bugnumber description) bug-alist))
      (when bugs-are-open-flag
       (when (and (re-search-forward
                    "Reported by: <a class=\\\"submitter\\\"
href=\\\"pkgreport.cgi\\?submitter=[^;]+;arch=source\\\">\"
                    nil t)
                  (or (looking-at "&quot;\\(.*\\)&quot; &lt;")
                      (looking-at "\\(.*\\) &lt;")))
        (setq shortdescription
         (concat "Bug fix: \" shortdescription
                  \"\", thanks to \"
                  (debian-bug-rfc2047-decode-string
                   (match-string 1))
                  \" (Closes: #\" bugnumber \").\")))
      (setq bug-open-alist
       (cons
        (list bugnumber shortdescription)
        bug-open-alist))))

```

実際に changelog に追加する部分は emacs-goodies-el:elisp/dpkg-dev-el/debian-changelog-mode.el(debian-changelog-close-bug) にあります .

7 次回



次回は 11 月 12 日土曜日の夜を予定しています．内容は小林さんによる DWN の翻訳フローについての話と，えとーさんによる dpkg-statoverride についての話しです．

参加者募集はまた後程．



Debian 勉強会資料

2005 年 10 月 29 日 初版第 1 刷発行

東京エリア Debian 勉強会（編集・印刷・発行）
