

# Debian GNU/Linux

Rob Funk

`rfunk@funknet.net`

`http://funknet.net/rfunk`

# History

- Started by Ian Murdock in 1993 in reaction to SLS
  - Failure of SLS's single-maintainer system
- Distributed-developer system from the start, modeled on kernel development
- "Debian" is combination of Ian's name with that of his wife Deborah
  
- Murdock left in 1996, Bruce Perens took over
- In 1997 Perens wrote Debian Social Contract, oriented Debian toward free software
- Perens' "Debian Free Software Guidelines" became the "Open Source Definition"
- Now Murdock back - commercial "Progeny"

# Philosophies (1)

## Open Development Model

- Active effort toward high quality
  - <http://bugs.debian.org/>
- Active mailing lists for all communication
  - <http://lists.debian.org/>
- Anyone may become a developer (currently ~700?)
  - Induction/mentorship process
- Decisions voted on among developers
  - <http://www.debian.org/devel/constitution>
- Policy is in publicly available documents
  - <http://www.debian.org/doc/debian-policy/>
- Strict standards compliance
  - e.g. Filesystem Hierarchy Standard

# Philosophies (2)

Careful attention to legalities and Free Software ideal

- "DFSG-Free" packages go in "main" section
  - Debian Free Software Guidelines became Open Source Definition
  
- Non-compliant packages available in separate "non-free" section
  - Free packages depending on "non-free" go in "contrib" section
  
- Cryptography goes in "non-US" to protect US mirrors
  - Due to US crypto export policy
  - Nothing wrong with using "non-US" packages in the US

# Technical Characteristics

- Many architectures supported
  - i386, alpha, powerpc, m68k, sparc, arm
  
- dpkg packaging system allows incremental upgrades without breakage
  
- Upgrade non-kernel packages without rebooting
  
- APT package tools enable very easy upgrading
  - Automatic package downloads
  - Automatic dependency updates - no dependency hell!
  - Put rpmfind out of business!
  - Best for those with fast network connections

# Three Variations

## unstable

- Bleeding-edge
- Latest of everything
- Often buggy

## testing

- Updated regularly (delayed from unstable)
- Nothing with known "release-critical" bugs
- Even security updates delayed for testing
- Compromise between most current and highest quality

## stable

- Best for production - reliable and high quality
- Not the latest software releases
- Only updated for security fixes and major bugs
- Release cycle approx. 1-2 years

Commercial versions such as Progeny and Libranet may be more current than "stable" but more stable than "testing".

# Release Naming

- Names correspond to specific release numbers
- Name sticks as the release moves through the unstable/testing/stable sequence
  - Except "sid" always unstable, never released
  - Name ("woody") and status ("testing") interchangeable in many contexts
- Debian 2.1 was slink, now obsolete
- Debian 2.2 is potato, now "stable"
  - Occasional update releases to stable
  - Now up to release 3 of 2.2 (2.2r3)
- woody will be Debian 2.3, currently "testing"
- After woody - maybe sarge?

Names taken from Toy Story movies (Bruce Perens worked for Pixar)

# Packaging System

- Package files are name\_version\_arch.deb
  
- dpkg - basic tool for install/remove/list
  - Mostly equivalent to rpm program
  - Many related dpkg-\* tools
  
- dselect - Visual available/installed package browser
  - Due to be replaced - not smart enough
  - Use apt as fetch method, for small improvement
  - Browse what's available, w/descriptions & categories
  - Presents Recommended/Suggested packages with selections
  
- APT - suite of "Advanced Package Tools"
  - apt-get - retrieve and install necessary packages
  - apt-cache - access database of available packages
  - aptitude & console-apt - dselect replacements?



# dpkg's Internal Install Process

- Extract control files
- If upgrading, run old version's 'prerm' script
- Run 'preinst' script
- Unpack new files
  - Backup old files if upgrading
- If upgrading, run old version's 'postrm' script
  
- Configure package
  - Unpack config files, backing up old ones
  - Run 'postinst' script
  - May ask questions of user - configurable threshold
  - May use update-\* programs to edit config files

# Handy update-\* scripts

- update-modules
  - Create modules.conf from files in /etc/modutils/
- update-rc.d
  - Edit init script links for runlevels
  - file-rc package uses configfile instead of links
- update-inetd
  - Edit inetd.conf
- update-menus
  - Consistent program menus in different window managers
- update-alternatives
  - Links in /etc/alternatives for default programs
  - e.g. vi -> vim or elvis or nvi

# .deb vs .rpm

## Recommendations

- Not enforced like dependencies
- Just says that the package will usually need this other package
- Often the other package depends on this one

## Suggestions

- Even looser than recommendations
- Says package sometimes works better if other one also installed

## Priorities

- required - can't run without (49 in potato)
- important - hard to run without (25 in potato)
- standard - generally expected (76 in potato)
- optional - possibly desired (3575 in potato)
- extra - may conflict with others (780 in potato)

## Standard Unix tools can create & extract

- ar, tar, gzip

## More extensible - new features easier to add

# From rpm to dpkg

List package(s)

`rpm -q pkgname`

`dpkg -l pkgpattern`

List all packages

`rpm -qa`

`dpkg -l`

Show long description of package

`rpm -qi pkgname`

`dpkg -p pkgname`

`dpkg -s pkgname`

Search for package owning file

`rpm -qf filename`

`dpkg -S filename`

List files owned by package

`rpm -ql pkgname`

`dpkg -L pkgname`

Install/Upgrade packages

`rpm -U pkgname`

`dpkg -i pkgname`

Purge package(s)

`rpm -e pkgname`

`dpkg -P pkgname`

Remove package(s), leaving config files

`---`

`dpkg -r pkgname`

apt-get often better than dpkg for installing/upgrading/removing

# APT - Advanced Package Tool

Configure list of repositories in `/etc/apt/sources.list`

- Get current contents of those repositories

`apt-get update`

- Access info about what's in the online repositories

`apt-cache search presentation`

`apt-cache show mgp`

- Install specific package(s) and their dependencies

`apt-get install mgp`

- Upgrade

- minor - packages with newer versions available

`apt-get upgrade`

- major - may remove some packages to install others

`apt-get dist-upgrade`

- Remove a package, and all that depend on it

`apt-get remove mgp`

# sources.list

```
# Specifying stable gets us potato until woody becomes stable,  
# then gets us woody as soon as it becomes stable.  
# Specify potato (or whatever release you choose)  
# if you want to be more careful about new releases.
```

```
deb http://ftp.us.debian.org/debian/ stable main non-free contrib  
deb-src http://ftp.us.debian.org/debian/ stable main non-free  
contrib
```

```
deb http://non-us.debian.org/debian-non-US stable/non-US main  
contrib non-free  
deb-src http://non-us.debian.org/debian-non-US stable/non-US main  
contrib non-free
```

```
deb http://security.debian.org stable/updates main contrib non-free
```

```
deb-src http://ftp.us.debian.org/debian/ woody main non-free  
contrib
```

```
deb http://pandora.debian.org/~robot101/mozilla /  
deb-src http://pandora.debian.org/~robot101/mozilla /
```

# Building Packages from Source

More complex than just a .spec file,  
but lots of tools make things easier

- Get and build all at once

```
apt-get --build source mgp
```

- Get, possibly modify then build

```
apt-get source mgp
```

```
cd mgp
```

```
dpkg-buildpackage
```

- Make a package out of a random source tree

```
tar xzf mgp.tar.gz
```

```
cd mgp
```

```
deb-make
```

```
dpkg-buildpackage
```

After building, you have a .deb file you can install with dpkg

# kernel-package

- Make package out of custom kernel
  - also make packages for headers and modules
  
- Simplifies kernel compilation
  - make-kpkg
  
- Eases management of custom kernels
  - upgrades won't mess you up
  - multiple kernels installed at once
  - multiple flavors of same release
  - allows patches as .deb packages

See `/usr/share/doc/kernel-package/RATIONALE.gz`



# When things go wrong...

- Sometimes there are errors between the unpacking stage and the configure stage

- List packages not correctly configured

```
dpkg -C
```

- Sometimes misc errors require fixing broken dependencies with:

```
apt-get install -f
```

- Sometimes packaging bugs require forcing

```
apt-get install --force-overwrite mqp
```

- See `dpkg --force-help`

- Bug database - <http://bugs.debian.org/>

- `reportbug`

Shouldn't be an issue in stable, but watch out in unstable and testing

# Production Tips

- Maybe just occasionally run apt-get commands:

```
apt-get update && apt-get dist-upgrade
```

```
sudo apt-get update && sudo apt-get dist-upgrade
```

- Automate: crontab entry to notify admin of updates

```
apt-get -qq update && apt-get -qq -s dist-upgrade
```

- Still need to do an `apt-get dist-upgrade` when you're notified

- Or automated upgrades, but that's dangerous

- netselect command tells nearest server given a list

# Installation

- Many media possible at various stages
  - CD, http/ftp, NFS, hard disk, floppies
  - Usually CD(s) or 3-5 floppies + net

## Major stages

- Boot and set basic parameters
  
- Install kernel and essential drivers
  - Necessities to get ~34MB base installation
  
- Get and extract base installation
  
- Boot base, then install extra packages with tasks  
(simple) or dselect (advanced)

# Installation Notes

- 34MB "base" more important than floppies
- Good idea to go into dselect once then immediately exit to let it install standard packages
- Don't get greedy with first run through dselect
  
- X config tool is "anXious"
- "sndconfig" ported from Red Hat into woody
  
- Only have to install once per machine
  - `apt-get dist-upgrade`
  
- Installation due to be overhauled after woody

# Documentation

- Package-specific docs

- /usr/doc/pkgname (potato) or /usr/share/doc/pkgname (woody)

- HOWTOs

- .../doc/HOWTO/en-txt/ and .../doc/HOWTO/en-html/
- "mini" under that

- Debian Documentation Project

- <http://www.debian.org/doc/ddp/>

- Dwarf's Guide To Debian GNU/Linux

- dwarfs-debian-guide package in woody
- <http://people.debian.org/psg/ddg/>

- Debian Planet

- <http://www.debianplanet.org/>

- Debian Help

- <http://www.debianhelp.org/>

- Mailing List Archives

- <http://lists.debian.org/>