

Debian Day PT 2008



Pacotes Debian

DebianDay  T
Agosto 2008

Carlos 'nifan' Lisboa



- Sistema de pacotes Debian

- O que é?

- Ferramentas disponíveis

- Dpkg
- Apt
- Aptitude
- Synaptic
- Tasksel
- Dselect

- Constituição dos pacotes



- Como fazer um pacote?
 - Do que necessitamos?
 - Primeiros passos
 - Ficheiros:
 - Changelog
 - Copyright
 - Control
 - Rules
 - Compilar o pacote

O que é?



- Servem para distribuir todos os ficheiros necessários a determinados comandos (programas) ou funcionalidades (bibliotecas)
- Apresentam-se com a extensão `.deb` ou `.udeb`
- São arquivos Unix `ar`

```
nifan@Osiris: ~
AR(1) GNU Development Tools

NAME
  ar - create, modify, and extract from archives

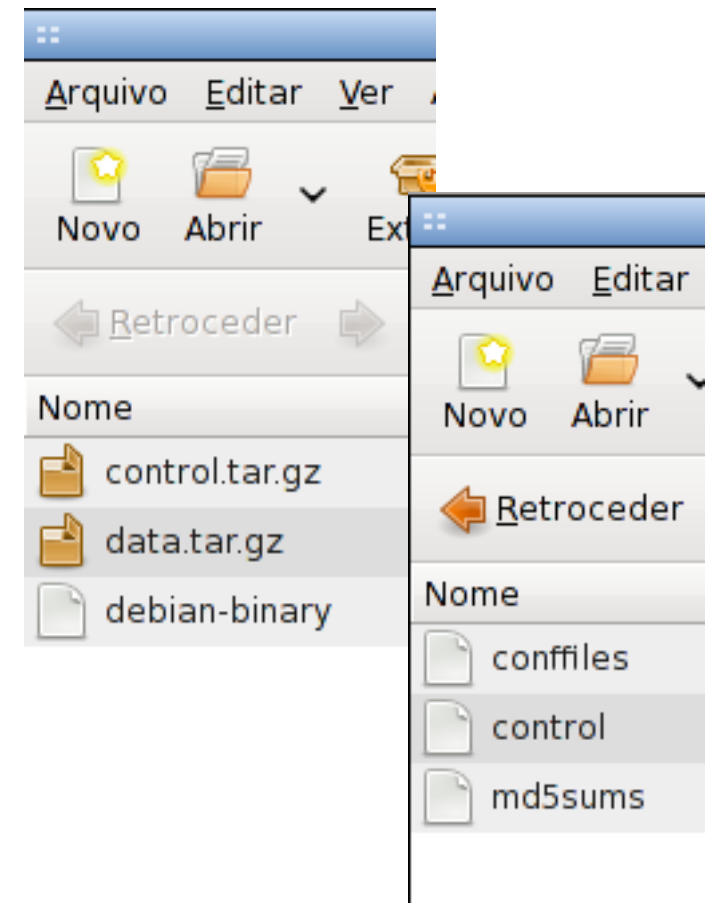
SYNOPSIS
  ar [-X32_64] [-]p[mod [relpos] [count]] archive [

DESCRIPTION
  The GNU ar program creates, modifies, and extra
  archive is a single file holding a collection o
```

O que é?



- Pacotes binários
 - Ficheiros executáveis
 - Ficheiros de configuração
 - Manpages
 - Outra documentação



O que é?



- Pacotes código-fonte
 - Código-fonte do programa/biblioteca que se pretende disponibilizar
 - Descrição do pacote

```
aptitude_0.4.11.9-1.dsc X
1 -----BEGIN PGP SIGNED MESSAGE-----
2 Hash: SHA1
3
4 Format: 1.0
5 Source: aptitude
6 Binary: aptitude, aptitude-dbg, aptitude
7 Architecture: any
8 Version: 0.4.11.9-1
9 Maintainer: Daniel Burrows <dburrows@de
10 Standards-Version: 3.8.0.0
11 Vcs-Browser: http://hg.debian.org/hg/ap
12 Vcs-Hg: http://hg.debian.org/hg/aptitude
13 Build-Depends: libapt-pkg-dev (>= 0.7.0
14 Checksums-Sha1:
```

O que é?



- Convencionalmente o nome de um pacote Debian deverá ter o seguinte formato:

nome_versão-revisãodebian_arquitectura.deb

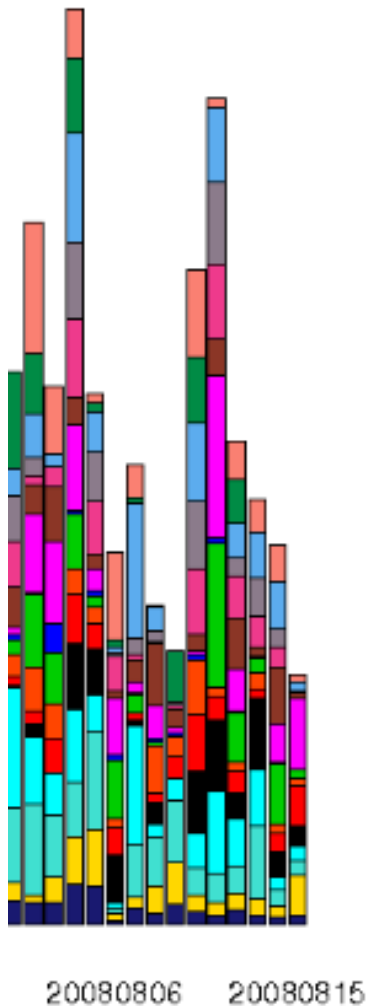
```
nifan@Osiris: /var/cache/apt/archives
nifan@Osiris: /var/cache/apt/archives$ ls -al aptitude_0.4.11.9-1_i386.deb
-rw-r--r-- 1 root root 2807856 Ago  4 02:17 aptitude_0.4.11.9-1_i386.deb
nifan@Osiris: /var/cache/apt/archives$
```

O que é?



- A construção de pacotes para a inclusão na distribuição segue regras e políticas rígidas, fazendo com que a qualidade dos pacotes seja a adequada.
 - Redistribuição livre
 - Código-fonte
 - Trabalho derivativo
 - Integridade do código-fonte do autor
 - Discriminação
 - Licenças
- [2.1 The Debian Free Soft](#)
 - [2.2 Categories](#)
 - [2.2.1 The main cate](#)
 - [2.2.2 The contrib ca](#)
 - [2.2.3 The non-free c](#)
 - [2.3 Copyright considerati](#)
 - [2.4 Sections](#)
 - [2.5 Priorities](#)
 - [3 Binary packages](#)
 - [3.1 The package name](#)
 - [3.2 The version of a pack](#)
 - [3.2.1 Version numbe](#)
 - [3.3 The maintainer of a p](#)
 - [3.4 The description of a p](#)

O que é?



- Existem aproximadamente 25 mil pacotes disponíveis nos repositórios Debian dispersos pelo main, contrib e non-free.
- Cada um dos pacotes é construído automaticamente por auto-builders para cada uma das arquiteturas.

Ferramentas Disponíveis



- Existem diversas ferramentas que podemos usar para gerir pacotes Debian.
- Dpkg – é a base, o apt utiliza-o para a gestão dos pacotes

```
nifan@Osiris: ~  
nifan@Osiris:~$ dpkg  
dpkg  
dpkg-architecture  
dpkg-buildpackage  
dpkg-checkbuilddeps  
dpkg-deb  
dpkg-depcheck  
dpkg-distaddfile  
dpkg-genbuilddeps  
dpkg-genchanges  
dpkg-gencontrol  
dpkg-gensymbols  
dpkg-name  
dpkg-parsechangelog  
dpkg-query  
dpkg-scanpackages  
dpkg-scansources  
dpkg-shlibdeps  
dpkg-source  
dpkg-split  
dpkg-trigger
```

```
nifan@Osiris: ~  
dpkg(1) dpkg suite  
NAME  
dpkg - package manager for Debian  
SYNOPSIS  
dpkg [options] action  
WARNING  
This manual is intended for users wishing  
line options and package states in more de  
dpkg --help.  
It should not be used by package maintain
```

Ferramentas Disponíveis



- Apt - resulta do nome Advanced Package Tool. Possui diversos comandos para manipular os pacotes, que vão desde a simples instalação/remoção de um pacote até à actualização de todo o sistema.

```
::
nifan@Osiris:~$ apt
apt-cache
apt-cdrom
apt-config
apt-extracttemplates
apt-ftparchive
apt-get
```

```
:: nifan@Osiris: ~
nifan@Osiris:~$ apt-cache search aptitude
aptitude - terminal-based package manager
aptitude-dbg - Debug symbols for the aptitude package
aptitude-doc-cs - Czech manual for aptitude, a terminal-based package manager
aptitude-doc-en - English manual for aptitude, a terminal-based package manager
aptitude-doc-fi - Finnish manual for aptitude, a terminal-based package manager
aptitude-doc-fr - French manual for aptitude, a terminal-based package manager
```



Apt-get demonstração

Ferramentas Disponíveis



- **Aptitude** - é um dos programas mais avançados para a gestão de pacotes. Possui a comum linha de comandos, bem como uma interface gráfica em ncurses.

```
:: nifan@Osiris: ~
APTITUDE(8) Command-Line Reference

NAME
  aptitude - high-level interface to the package manager

SYNOPSIS
  aptitude [options...] {autoclean
  update | safe-upgrade}

  aptitude [options...] {changelog
  forbid-version | hold | i
  | reinstall | remove | sh
  | build-depends} <package>
```

```
:: nifan@Osiris: ~
Acções Desfazer Pacote Resolver Procurar Opções V
C-T: Menu ?: Ajuda q: Sair u: Actualizar g: Downloa
aptitude 0.4.11.9 Serão usados 188kB de
-- Pacotes Actualizáveis (42)
-- Novos Pacotes (1243)
-- Pacotes Instalados (1483)
-- Pacotes não Instalados (21591)
-- Pacotes Obsoletos e Criados Localmente (17)
-- Pacotes Virtuais (2532)
-- Tarefas (717)

Está disponível uma versão mais recente destes pacotes.
Este grupo contém 42 pacotes.
```

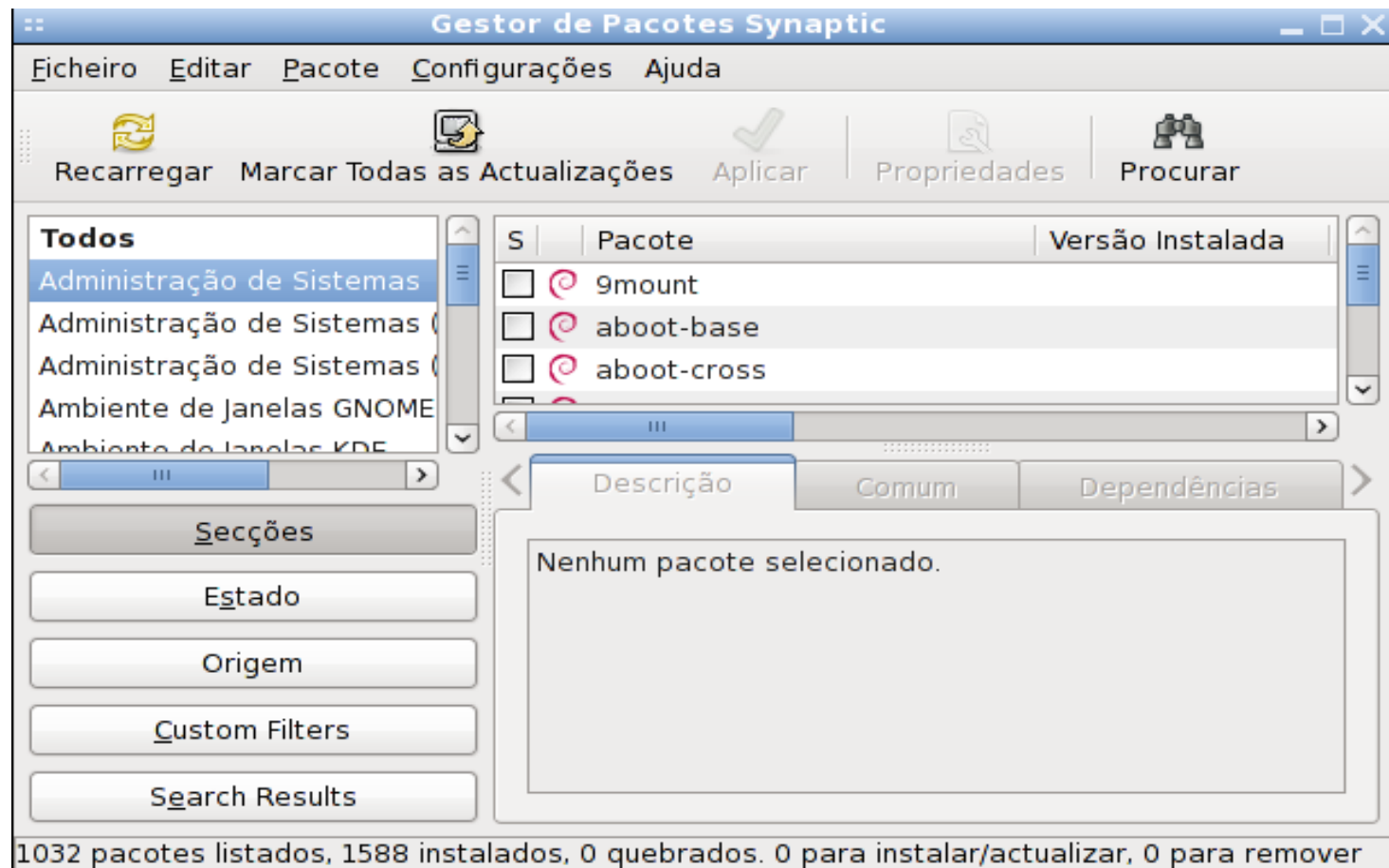


Aptitude demonstração

Ferramentas Disponíveis



- Synaptic – é o gestor de pacotes gráfico por excelência. Permite uma gestão mais simples.



Ferramentas Disponíveis



- **Tasksel** – é usado no Debian Installer e permite a instalação de grupos de pacotes por função (Desktop, Servidor, etc.)

```
nifan@Osiris: ~  
Configuração de pacotes  
----- Seleção de software -----  
Pode escolher instalar uma ou mais das seguintes  
pré-definidas.  
  
Escolha o software a instalar:  
  
[*] Ambiente de Desktop  
[ ] Servidor Web  
[*] Servidor de Impressão  
[ ] Servidor DNS  
[ ] Servidor de ficheiros  
[ ] Servidor de Mail  
[ ] Base de dados SQL  
[*] Portátil  
[ ] selecção manual de pacotes
```

Ferramentas Disponíveis



- **Dselect** – foi o precursor do aptitude em versões mais antigas Debian. Actualmente os utilizadores são encorajados a utilizar o aptitude.

```
:: nifan@Osiris: ~
Frontend de gestão de pacotes Debian `dselect' versão 1.14.20 (i386)

* 0. [A]cesso      Escolha o método a utilizar.
  1. Act[U]aliza   Actualizar a lista de pacotes disponíveis, se possi
  2. [S]eleccion  Solicite quais pacotes quer no seu sistema.
  3. [I]nstalar   Instalar e actualizar os pacotes desejados.
  4. [C]onfig     Configura pacotes ainda não configurados.
  5. [R]emover    Remover software não desejado
  6. [Q] Sair     Sair do dselect.

Mova-se com ^P e ^N, teclas de cursor, letras iniciais, ou dígitos;
Carregue em <enter> para confirmar a selecção.  ^L re-desenha o ecrã
Copyright (C) 1994-1996 Ian Jackson.
Copyright (C) 2000,2001 Wichert Akkerman.
```

Ferramentas Disponíveis



- Outras ferramentas:
 - Apt-build – compilar pacotes
 - Apt-proxy – proxy de pacotes para diminuir tráfego
 - Apt-move – mover ficheiros dentro de um repositório
 - Apt-file – lista de ficheiros de pacotes
 - Apt-cache – listagem de pacotes e suas informações
 - Apt-listchanges – mostra alterações nos pacotes
 - Netselect-apt – seleccionar o servidor mais rápido

Ferramentas Disponíveis



- Outras ferramentas:
 - Apt-listbugs – bugs abertos num determinado pacote
 - Apt-p2p – permite a partilha de pacotes entre utilizadores
 - Debtorrent – distribuição de pacotes através de torrents
 - Dpkg-reconfigure – configuração de pacotes com debconf
 - Debtorphan – listagem de pacotes “orfãos”
 - Apt-get remove --purge `deborphan` - mesmo que anterior

Constituição dos pacotes



- Pacotes binários:

- control.tar.gz

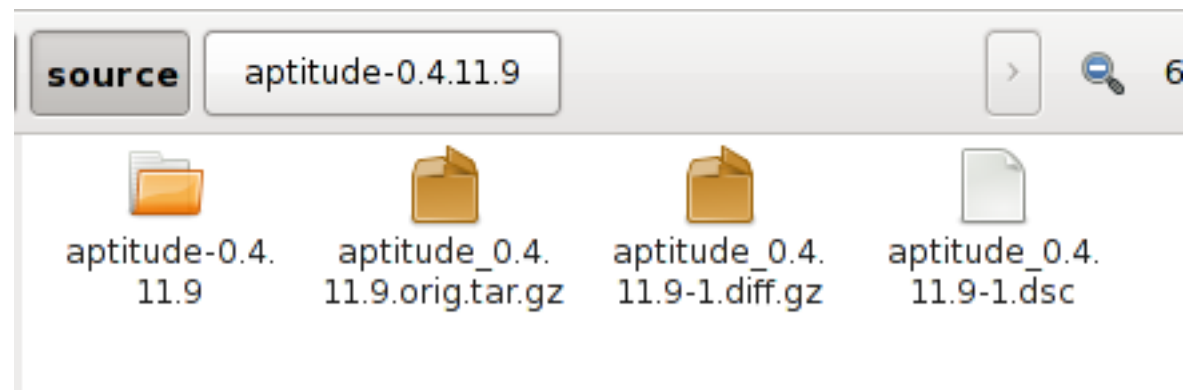
- Conffile
- Preinst
- Postinst
- Prerm
- Postrm
- md5sums

```
1 |abb3978ccbd172920d8c00af143017fd  usr/share/doc/aptitude/TODO
2 |5580746074c22a6945a77204c85a3e90  usr/share/doc/aptitude/README.hie
3 |0b182179bd205a95a1d1fb01145c4441  usr/share/doc/aptitude/README.Deb
4 |9e26c1d35b2f2334608c5a3828290906  usr/share/doc/aptitude/copyright
5 |af22003d389ad29b149c54888a8017c3  usr/share/doc/aptitude/examples/p
```

Constituição dos pacotes



- Pacotes código-fonte
 - .dsc
 - orig.tar.gz
 - diff.gz



Constituição dos pacotes



- Cada pacote possui uma determinada prioridade:
 - **Required**: essenciais ao funcionamento do sistema
 - **Important**: constituem uma infraestrutura básica
 - **Standard**: pacotes padrão em qualquer sistema Linux (browser, email, cliente ftp)
 - **Optional**: pacotes com programas que os utilizadores desejem utilizar
 - **Extra**: pacotes para funcionalidades específicas (pode haver conflito com pacotes com prioridades mais elevadas)

```
.. nifan@Osi
nifan@Osisiris:~$ dpkg --status aptitude
Package: aptitude
Status: install ok installed
Priority: important
Section: admin
```

Constituição dos pacotes



- Podem haver pacotes especiais que servem apenas como recipiente para um grupo específico de pacotes, são os chamados pacotes virtuais.
- Quando mais que um pacote fornece uma determinada funcionalidade, pode ser escolhido qual o programa padrão com o `update-alternatives`.

```
p vimnetp-de - vi Improved - Documentation FI
nifan@osiris:~$ aptitude show gvim
Nenhuma foi encontrada nenhuma versão actual ou candidata para gvim
Pacote: gvim
Estado: não é um pacote verdadeiro
Disponibilizado por: vim-gnome, vim-gtk, vim-lesstif
nifan@osiris:~$
```

Constituição dos pacotes



- Dependências entre pacotes:
 - Depends
 - Recommends
 - Suggests
 - Conflicts
 - Replaces
 - Provides

```
:: nifan@Osiris: ~  
nifan@Osiris:~$ aptitude why aptitude  
i  tasksel Depende aptitude (>= 0.2.15-1)  
nifan@Osiris:~$ aptitude why-not textopo  
i  w3m Sugere w3m-el  
p  w3m-el Sugere wv  
p  wv Sugere tetex-extra  
p  tetex-extra Depende texlive-latex-extra (>= 2007)  
p  texlive-latex-extra Em Conflito textopo
```

Constituição dos pacotes



- Acções determinadas pelo utilizador
 - Install
 - Remove
 - Purge
 - Hold

```

:: nifan@Osiris: ~
nifan@Osiris:~$ dpkg --get-selections > pacotes.txt
nifan@Osiris:~$ head pacotes.txt
accerciser                install
acpi                      install
acpi-support              install
acpi-support-base        install
acpid                    install
adduser                  install
adobereader-enu          install

```

Constituição dos pacotes



- Podemos obter o código fonte de um determinado pacote para modificarmos o que quisermos.
- Cada pacote possui dependências de compilação, isto é, são necessários determinados pacotes para que a sua construção correcta

```
nifan@Osiris:~$ sudo apt-get build-dep aptitude
[sudo] password for nifan:
A ler as listas de pacotes... Pronto
A construir árvore de dependências
A ler a informação de estado... Pronto
Serão instalados os seguintes NOVOS pacotes:
 libapt-pkg-dev libcppunit-1.12-1 libcppunit-dev
 libncursesw5-dev libsgmls-perl libsigc++-2.0-dev
```

Como Fazer um Pacote?



- Do que necessitamos?

- Para a compilação de um determinado pacote necessitamos de alguns programas essenciais na construção de todos os pacotes debian.

- Dpkg-dev – contém diversos programas para construir pacotes Debian.
- Make – normalmente os programas necessitam de diversos passos para a sua compilação, estes passos são agrupados em Makefiles.
- Patch – útil para produzir e aplicar alterações em ficheiros
- Gcc – compilador e utilidades

```
:: nifan@Osiris: ~
nifan@Osiris:~$ sudo aptitude install build-essential
[sudo] password for nifan:
A ler as listas de pacotes... Pronto
A construir árvore de dependências
A ler a informação de estado... Pronto
A ler informações extendidas de estado
A inicializar os estados dos pacotes... Pronto
A ler as descrições das tarefas... Pronto
```

Como Fazer um Pacote?



- Para além dos programas anteriores devermos ter:
 - Autoconf e Automake – necessários para alguns programas com Makefiles pré-processados
 - dh_make e debhelper – facilita o processo de construção do pacote
 - fakeroot – cria um ambiente root emulado
 - Lintian – verificação do pacote criado
 - Pbuilder – verificação de um pacote em chroot para verificar falhas na instalação/remoção/actualização

```
:: nifan@Osiris: ~
nifan@Osiris:~$ lintian aptitude_0.4.11.9-1_i386.deb
W: aptitude: manpage-has-errors-from-man usr/share/man/ja/man8/aptitude
: warning: `Sh' not defined
E: aptitude: python-script-but-no-python-dep ./usr/share/bug/aptitude
```

Primeiros Passos



- Obter o programa
- Colocar o código-fonte do programa num directório com o nome no formato programa-versão.
- Debianizar o programa com:
 - `dh_make -e nome@isp.pt -f ../programa_versao.tar.gz`

Isto irá perguntar se pretende criar um pacote simples, múltiplo, biblioteca. Após o qual criar um directório debian com diversos ficheiros.

```
:: nifan@Osiris: ~/debian
nifan@Osiris:~/debian$ ls
changelog  emacsen-install.ex  manpage.xml.ex  rules
compat    emacsen-remove.ex  menu.ex
control   emacsen-startup.ex  postinst.ex
copyright  init.d.ex           postrm.ex
cron.d.ex  init.d.lsb.ex       preinst.ex
dirs      manpage.1.ex        prerm.ex
docs      manpage.sgml.ex     README.Debian
```

Primeiros Passos



- Com o último passo foi criado um ficheiro com o nome `programa_versão.orig.tar.gz` que contém o código fonte tal como o autor o escreveu. As alterações que forem efectuadas no código fonte serão compiladas num `diff.gz`.
- Por vezes o Makefile de um determinado programa pode ter os caminhos colocados directamente.

-BIN = `/usr/bin`

Deverá ser mudado para:

-BIN = `$(DESTDIR)/usr/bin`

Changelog



- Neste ficheiro serão colocadas informações sobre as alterações e correcções importantes de cada versão e o seu autor.

```
changelog x
1 programa (versão-revisão) distribuição; urgency=low
2
3 * Initial release (Closes: #nnnn) <nnnn is the bug number of your ITP>
4
5 -- Autor <email@isp.pt> Sat, 16 Aug 2008 16:00:00 +0100
6
```

Copyright



- Neste ficheiro serão colocadas informações como o autor do código fonte, a licença com a qual é distribuído o código e quem “debianizou” o programa.

```
copyright x
1 This package was debianized by autor <email> on
2 Sat, 16 Aug 2008 18:00:00 +0100.
3
4 It was downloaded from <url://example.com>
5
6 Upstream Author(s):
7
8     <put author's name and email here>
9     <likewise for another author>
10
11 Copyright:
12
13     <Copyright (C) YYYY Name OfAuthor>
14     <likewise for another author>
15
16 License:
17
18     <Put the license of the package here indented by 4 spaces>
19
```



- Este é um dos ficheiros mais importantes. É nele que são definidas as dependências e as dependências necessárias à construção do pacote. Possui ainda uma descrição do pacote.

```
control x
1 Source: programa
2 Section: unknown
3 Priority: extra
4 Maintainer: Autor <email>
5 Build-Depends: debhelper (>= 7), autotools-dev
6 Standards-Version: versão
7 Homepage: <insert the upstream URL, if relevant>
8
9 Package: programa
10 Architecture: any
11 Depends: ${shlibs:Depends}, ${misc:Depends}
12 Description: <insert up to 60 chars description here>
13 <insert long description, indented with space>
```

Control



- O dh_make colocou no ficheiro control algo como:

– `-${shlibs:Depends}`

Isto não é mais que uma chamada para o programa dh_shlibdeps que irá compilar a lista de dependências das bibliotecas partilhadas. Esta lista será passada para o dh_gencontrol que irá colocar no devido local.

- Uma forma de obter a lista de dependências de compilação (Build-Depends) é com:

– `Objdump -p binário | grep NEEDED`

Seguido de `dpkg -S biblioteca` para determinar as bibliotecas `-dev` necessárias.

rules



- Este ficheiro não é mais que um Makefile.
- Para programas simples não deverá ser necessário alterar nada, o template colocado pelo dh_make deverá ser suficiente.

```
rules x
35
36
37 build: build-stamp
38
39 build-stamp: config.status
40     dh_testdir
41
42     # Add here commands to compile the
43     $(MAKE)
44     #docbook-to-man debian/tcpdump.sgml
45
46     touch $@
47
48 clean:
49     dh_testdir
50     dh_testroot
```

Rules



- Neste ficheiro podemos encontrar referência a diversos programas `dh_*`. Estes programas possuem funções específicas e podem ser comentadas/descomentadas conforme for necessário. Os mais comuns e úteis serão:
 - `dh_strip`- remove informação de debug tornando o binário mais pequeno.
 - `dh_shlibdeps` – já mencionado, cria a lista de dependências.
 - `dh_gencontrol` – criação do control
 - `dh_md5sums` – devolve as hash's md5 dos ficheiros do pacote.

Compilar o Pacote



- Chegamos à parte final, a construção em si do pacote. É tão simples quanto:
 - `dpkg-buildpackage -rfakeroot`
- Após este passo estarão diversos ficheiros no directório acima do pacote.
 - `programa_versao-revisão.orig.tar.gz`
 - `programa_versao-revisão.diff.gz`
 - `programa_versao-revisão_arquitectura.deb`
 - `programa_versao-revisão.changes`



Questões?

Referências



- <http://mentors.debian.net>
- <http://www.debian.org/doc/debian-policy/>
- <http://www.debian.org/doc/maint-guide/>
- <http://video.google.com/videoplay?docid=-6726522426109060914>
- <http://wiki.debian.org/HowToPackageForDebian>
- <http://www.debian.org/distrib/packages>
- http://www.debian.org/doc/FAQ/ch-pkg_basics.en.html
- http://tldp.org/HOWTO/html_single/Debian-Binary-Package-Building-HOWTO/
- http://www.debian.org/social_contract#guidelines