

Debian Packaging Tutorial

Magic that makes `"apt-get install"` work

Muneeb Shaikh AbdulKarim Memon

February 26, 2012



Outline

- 1 General Installation Procedure
- 2 Packaging
- 3 Creating Debian Package Steps
- 4 References



Outline

- 1 General Installation Procedure
 - From Source
 - From Repository
- 2 Packaging
 - Tools of Trade
 - General packaging workflow
- 3 Creating Debian Package Steps
 - Setting up your BASH environment
 - Download upstream tarball
 - Rename the upstream tarball
 - Unpack the upstream tarball
 - Add the Debian packaging files
 - Files in debian/
 - Build the package
 - Check for errors
 - Install the package
 - Uploading to mentors.debian.net
- 4 References



From Source

- ▶ Download the source from upstream
- ▶ Read the installation instructions
- ▶ Hunt for the pre-requisites of installing (Download Dependencies)
- ▶ Finally install with these commands
 - ❶ `./configure`
 - ❷ `make`
 - ❸ `make install`



Outline

- 1 General Installation Procedure
 - From Source
 - From Repository
- 2 Packaging
 - Tools of Trade
 - General packaging workflow
- 3 Creating Debian Package Steps
 - Setting up your BASH environment
 - Download upstream tarball
 - Rename the upstream tarball
 - Unpack the upstream tarball
 - Add the Debian packaging files
 - Files in debian/
 - Build the package
 - Check for errors
 - Install the package
 - Uploading to mentors.debian.net
- 4 References



From Repository

```
sudo apt-get install package_name
```



Outline

- 1 General Installation Procedure
- 2 Packaging
- 3 Creating Debian Package Steps
- 4 References



Outline

- 1 General Installation Procedure
 - From Source
 - From Repository
- 2 Packaging
 - Tools of Trade
 - General packaging workflow
- 3 Creating Debian Package Steps
 - Setting up your BASH environment
 - Download upstream tarball
 - Rename the upstream tarball
 - Unpack the upstream tarball
 - Add the Debian packaging files
 - Files in debian/
 - Build the package
 - Check for errors
 - Install the package
 - Uploading to mentors.debian.net
- 4 References



Tools of Trade

- ▶ A Debian (or Ubuntu) system (with root access)
- ▶ Some packages:
 - ▶ **build-essential**: contains basic building tools such as **gcc**, **g++**, **make** and mainly **dpkg-dev**, which contains basic Debian-specific tools to create packages
 - ▶ **devscripts**: contains many useful scripts for Debian maintainers
 - ▶ **dh-make**: tool to Debianize the upstream source easily
 - ▶ **lintian**: Debian package checker

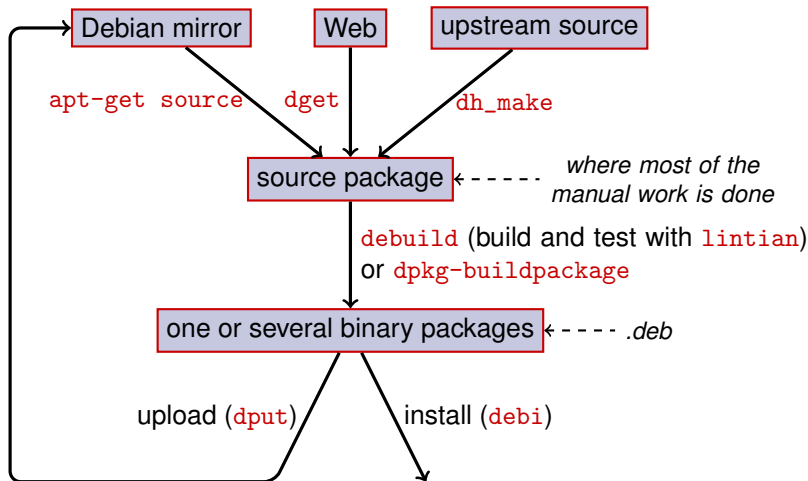


Outline

- 1 General Installation Procedure
 - From Source
 - From Repository
- 2 Packaging
 - Tools of Trade
 - General packaging workflow
- 3 Creating Debian Package Steps
 - Setting up your BASH environment
 - Download upstream tarball
 - Rename the upstream tarball
 - Unpack the upstream tarball
 - Add the Debian packaging files
 - Files in debian/
 - Build the package
 - Check for errors
 - Install the package
 - Uploading to mentors.debian.net
- 4 References



General packaging workflow



Outline

- 1 General Installation Procedure
- 2 Packaging
- 3 Creating Debian Package Steps
- 4 References



Creating Debian Package Steps

- ➊ Setting up your BASH environment
- ➋ Download the upstream tarball
- ➌ Rename the upstream tarball
- ➍ Unpack the upstream tarball
- ➎ Add the Debian packaging files
- ➏ Build the package
- ➐ Check for errors
- ➑ Install the package
- ➒ If everything is working as expected upload it to mentors.debian.net



Outline

- 1 General Installation Procedure
 - From Source
 - From Repository
- 2 Packaging
 - Tools of Trade
 - General packaging workflow
- 3 Creating Debian Package Steps
 - Setting up your BASH environment
 - Download upstream tarball
 - Rename the upstream tarball
 - Unpack the upstream tarball
 - Add the Debian packaging files
 - Files in debian/
 - Build the package
 - Check for errors
 - Install the package
 - Uploading to mentors.debian.net
- 4 References



Step 1: Setting up your BASH environment

- ▶ Append the following lines to `~/.bashrc`
[replace the values with Your full name and email address]

```
DEBEMAIL="abdulkarimmemon@gmail.com"  
DEBFULLNAME="AbdulKarim Memon"  
export DEBFULLNAME DEBEMAIL
```

- ▶ Restart your terminal or execute following in the terminal.

```
$ source ~/.bashrc
```



Outline

- 1 General Installation Procedure
 - From Source
 - From Repository
- 2 Packaging
 - Tools of Trade
 - General packaging workflow
- 3 Creating Debian Package Steps
 - Setting up your BASH environment
 - Download upstream tarball**
 - Rename the upstream tarball
 - Unpack the upstream tarball
 - Add the Debian packaging files
 - Files in debian/
 - Build the package
 - Check for errors
 - Install the package
 - Uploading to mentors.debian.net
- 4 References



Step 2: Download upstream tarball

- 1 Create a new directory so that we can work in clean environment
- 2 Download the source package.

```
$ mkdir ~/packaging
$ cd ~/packaging
$ wget http://download.savannah.gnu.org/releases/smc/
  hyphenation/patterns/hyphen-as-0.7.0.tar.bz2
```



Outline

- 1 General Installation Procedure
 - From Source
 - From Repository
- 2 Packaging
 - Tools of Trade
 - General packaging workflow
- 3 **Creating Debian Package Steps**
 - Setting up your BASH environment
 - Download upstream tarball
 - Rename the upstream tarball**
 - Unpack the upstream tarball
 - Add the Debian packaging files
 - Files in debian/
 - Build the package
 - Check for errors
 - Install the package
 - Uploading to mentors.debian.net
- 4 References



Step 3: Rename the upstream tarball

Rename the tarball to `<package_name>_<version>.orig.tar.bz2`

```
$ mv hyphen-as-0.7.0.tar.bz2 hyphen-as_0.7.0.orig.tar.  
bz2
```



Outline

- 1 General Installation Procedure
 - From Source
 - From Repository
- 2 Packaging
 - Tools of Trade
 - General packaging workflow
- 3 Creating Debian Package Steps
 - Setting up your BASH environment
 - Download upstream tarball
 - Rename the upstream tarball
 - Unpack the upstream tarball**
 - Add the Debian packaging files
 - Files in debian/
 - Build the package
 - Check for errors
 - Install the package
 - Uploading to mentors.debian.net
- 4 References



Step 4: Unpack the upstream tarball

```
$ tar xvf hyphen-as_0.7.0.orig.tar.bz2
hyphen-as-0.7.0/
hyphen-as-0.7.0/hyph_as_IN.dic
hyphen-as-0.7.0/README
hyphen-as-0.7.0/ChangeLog
hyphen-as-0.7.0/openoffice.org-hyphenation-as
hyphen-as-0.7.0/COPYING
hyphen-as-0.7.0/Makefile
```



Outline

- 1 General Installation Procedure
 - From Source
 - From Repository
- 2 Packaging
 - Tools of Trade
 - General packaging workflow
- 3 **Creating Debian Package Steps**
 - Setting up your BASH environment
 - Download upstream tarball
 - Rename the upstream tarball
 - Unpack the upstream tarball
 - Add the Debian packaging files**
 - Files in debian/
 - Build the package
 - Check for errors
 - Install the package
 - Uploading to mentors.debian.net
- 4 References



Step 5: Add the Debian packaging files

- 1 Change to the extracted source.

```
$ cd hyphen-as-0.7.0
```

- 2 To add debian related files execute following command.

```
$ dh_make -c gpl3
```

```
$ dh_make -c gpl3
```

```
Type of package: single binary, indep binary, multiple binary, library, kernel module, kernel patch?  
[s/i/m/l/k/n] s
```

```
Maintainer name   : Muneeb Shaikh  
Email-Address     : iamuneeb@gmail.com  
Date              : Sun, 26 Feb 2012 03:38:41 +0530  
Package Name      : hyphen-as  
Version           : 0.7.0  
License           : gpl3  
Type of Package   : Single
```

```
Hit <enter> to confirm:
```

```
Skipping creating ../hyphen-as_0.7.0.orig.tar.bz2 because it already exists  
Done. Please edit the files in the debian/ subdirectory now. You should also  
check that the hyphen-as Makefiles install into $DESTDIR and not in / .
```



Outline

- 1 General Installation Procedure
 - From Source
 - From Repository
- 2 Packaging
 - Tools of Trade
 - General packaging workflow
- 3 **Creating Debian Package Steps**
 - Setting up your BASH environment
 - Download upstream tarball
 - Rename the upstream tarball
 - Unpack the upstream tarball
 - Add the Debian packaging files
 - Files in debian/**
 - Build the package
 - Check for errors
 - Install the package
 - Uploading to mentors.debian.net
- 4 References



Files in debian/

All the packaging work should be made by modifying files in `debian/`

- ▶ Main files:
 - ▶ **control** – meta-data about the package (dependencies, etc)
 - ▶ **rules** – specifies how to build the package
 - ▶ **copyright** – copyright information for the package
 - ▶ **changelog** – history of the Debian package
- ▶ Other files:
 - ▶ `compat`
 - ▶ `watch`
 - ▶ `dh_install*` targets
 - `*.dirs`, `*.docs`, `*.manpages`, ...
 - ▶ maintainer scripts
 - `*.postinst`, `*.prerm`, ...
 - ▶ `source/format`
 - ▶ `patches/` – if you need to modify the upstream sources
- ▶ Several files use a format based on RFC 822 (mail headers)



debian/changelog

- ▶ Lists the Debian packaging changes
- ▶ Gives the current version of the package

1.2.1.1-5
Upstream Debian
version revision

- ▶ Edited manually or with `dch`
- ▶ Special format to automatically close Debian or Ubuntu bugs
Debian: Closes: #595268; Ubuntu: LP: #616929
- ▶ Installed as `/usr/share/doc/package/changelog.Debian.gz`

```
hyphen-hi (0.6.0-1) unstable; urgency=low
```

```
* Initial release (Closes: #542240)
```

```
-- Muneeb Shaikh <iammuneeb@gmail.com> Sun, 31 Jul 2011 18:11:17 +05
```



debian/control

- ▶ Package metadata
 - ▶ For the source package itself
 - ▶ For each binary package built from this source
- ▶ Package name, section, priority, maintainer, uploaders, build-dependencies, dependencies, description, homepage, ...
- ▶ Documentation: Debian Policy chapter 5
<http://www.debian.org/doc/debian-policy/ch-controlfields.html>



Sample debian/control

```
Source: hyphen-hi
Section: text
Priority: optional
Maintainer: Muneeb Shaikh <iammuneeb@gmail.com>
Uploaders: Debian-IN Team <debian-in-workers@lists.alioth.debian.org>
Build-Depends: debhelper (>= 7.0.50~),
               dictionaries-common (>= 0.10)
Standards-Version: 3.9.2
Homepage: http://wiki.smc.org.in/Hyphenation
Vcs-Git: git://git.debian.org/debian-in/hyphen-hi.git
Vcs-Browser: http://git.debian.org/?p=debian-in/hyphen-hi.git;a=summary
```

```
Package: hyphen-hi
Architecture: all
Depends: ${misc:Depends},
        dictionaries-common (>= 0.10)
Recommends: libreoffice-writer | openoffice.org-writer
Description: Hindi hyphenation patterns for OpenOffice.org/LibreOffice
Hyphenation is the process of inserting hyphens in between the syllables of
a word so that when the text is justified, maximum space is utilized.
```

This package provides the hyphenation rules for Hindi language.



Architecture: all or any

Two kinds of binary packages:

- ▶ Packages with different contents on each Debian architecture
 - ▶ Example: C program
 - ▶ Architecture: `any` in `debian/control`
 - ▶ Or, if it only works on a subset of architectures:
`Architecture: amd64 i386 ia64 hurd-i386`
 - ▶ `buildd.debian.org`: builds all the other architectures for you on upload
 - ▶ Named `package_version_architecture.deb`
- ▶ Packages with the same content on all architectures
 - ▶ Example: Perl library
 - ▶ Architecture: `all` in `debian/control`
 - ▶ Named `package_version_all.deb`

A source package can generate a mix of Architecture: `any` and
Architecture: `all` binary packages



debian/rules

- ▶ Makefile
- ▶ Interface used to build Debian packages
- ▶ Documented in Debian Policy, chapter 4.8
<http://www.debian.org/doc/debian-policy/ch-source.html#s-debianrules>
- ▶ Five required targets:
 - ▶ `build`: should perform all the configuration and compilation
 - ▶ `binary`, `binary-arch`, `binary-indep`: build the binary packages
 - ▶ `dpkg-buildpackage` will call `binary` to build all the packages, or `binary-arch` to build only the `Architecture: any` packages
 - ▶ `clean`: clean up the source directory



Outline

- 1 General Installation Procedure
 - From Source
 - From Repository
- 2 Packaging
 - Tools of Trade
 - General packaging workflow
- 3 **Creating Debian Package Steps**
 - Setting up your BASH environment
 - Download upstream tarball
 - Rename the upstream tarball
 - Unpack the upstream tarball
 - Add the Debian packaging files
 - Files in debian/
 - Build the package**
 - Check for errors
 - Install the package
 - Uploading to mentors.debian.net
- 4 References



Step 6: Build the package

- ▶ Various tools to build the package.
 - ▶ **dbkg-buildpackage**
 - ▶ **debuild**
 - ▶ **pdebuild**
 - ▶ **git-buildpackage**
- ▶ Execute following command to build the package.
\$ dpkg-buildpackage -us -uc

```
dpkg-buildpackage: export CFLAGS from dpkg-buildflags (origin: vendor): -g -O2
dpkg-buildpackage: export CPPFLAGS from dpkg-buildflags (origin: vendor):
.
.
.
dpkg-deb: building package 'hyphen-hi' in '../hyphen-hi_0.7.0-2_all.deb'.
dpkg-genchanges >../hyphen-hi_0.7.0-2_amd64.changes
dpkg-genchanges: not including original source code in upload
dpkg-source --after-build hyphen-hi
dpkg-buildpackage: binary and diff upload (original source NOT included)
```



Outline

- 1 General Installation Procedure
 - From Source
 - From Repository
- 2 Packaging
 - Tools of Trade
 - General packaging workflow
- 3 **Creating Debian Package Steps**
 - Setting up your BASH environment
 - Download upstream tarball
 - Rename the upstream tarball
 - Unpack the upstream tarball
 - Add the Debian packaging files
 - Files in debian/
 - Build the package
 - Check for errors**
 - Install the package
 - Uploading to mentors.debian.net
- 4 References



Step 6: Check for errors

- Use **lintian** to check the errors and warnings in package.

```
$ lintian hyphen-hi_0.7.0-2_amd64.changes
```

```
W: hyphen-hi source: no-debian-copyright
```

```
E: hyphen-hi: no-copyright-file
```



Outline

- 1 General Installation Procedure
 - From Source
 - From Repository
- 2 Packaging
 - Tools of Trade
 - General packaging workflow
- 3 **Creating Debian Package Steps**
 - Setting up your BASH environment
 - Download upstream tarball
 - Rename the upstream tarball
 - Unpack the upstream tarball
 - Add the Debian packaging files
 - Files in debian/
 - Build the package
 - Check for errors
 - Install the package**
 - Uploading to mentors.debian.net
- 4 References



Step 7: Install the package

- Use **dpkg** to install the package.

```
$ sudo dpkg -i hyphen-hi_0.7.0-2_all.deb
```

```
[sudo] password for muneeb:
Selecting previously deselected package hyphen-hi.
(Reading database ... 602891 files and directories currently installed)
Unpacking hyphen-hi (from hyphen-hi_0.7.0-2_all.deb) ...
Setting up hyphen-hi (0.7.0-2) ...
Processing triggers for postgresql-common ...
Building PostgreSQL dictionaries from installed myspell/hunspell packages
en_au
en_ca
en_gb
en_us
en_za
```



Outline

- 1 General Installation Procedure
 - From Source
 - From Repository
- 2 Packaging
 - Tools of Trade
 - General packaging workflow
- 3 Creating Debian Package Steps
 - Setting up your BASH environment
 - Download upstream tarball
 - Rename the upstream tarball
 - Unpack the upstream tarball
 - Add the Debian packaging files
 - Files in debian/
 - Build the package
 - Check for errors
 - Install the package
 - Uploading to mentors.debian.net
- 4 References



Step 8: Uploading to mentors.debian.net

- ▶ About `mentors.debian.net`
- ▶ Catch us on IRC or via email to know more about configuring it
 - ▶ IRC: **#debian-in-mentors** @ irc.oftc.net
 - ▶ Mailing List: `debian-in-workers`
- ▶ Upload the package using `dput`
`$ dput mentors-ftp hyphen-hi_0.7.0-2_amd64.changes`



Outline

- 1 General Installation Procedure
- 2 Packaging
- 3 Creating Debian Package Steps
- 4 References



References

- ▶ Debian New Maintainers' Guide
<http://www.debian.org/doc/manuals/maint-guide/>
- ▶ Packaging Tutorial
<http://wiki.debian.org/PackagingTutorial>
- ▶ Intro to Debian Packaging
<http://wiki.debian.org/IntroDebianPackaging>¹



Queries?



Queries?
Thank You

