Debian Med packaging workshop

Package your very own medical application

Andreas Tille

Debian

Required knowledge

- **makefile**
- diff / patch (quilt)
- SVN or Git
- debhelper
- lintian
Required knowledge

- makefile
- diff / patch (quilt)
- SVN or Git
- debhelper
- lintian
Required knowledge

- makefile
- diff / patch (quilt)
- SVN or Git
- debhelper
- lintian
Required knowledge

- makefile
- diff / patch (quilt)
- SVN or Git
- debhelper
- lintian
Short Introduction

Let's start with your own example

- makefile
- diff / patch (quilt)
- SVN or Git
- debhelper
- lintian
Do-O-cracy
Debian Med policy
Debian Med
packaging
workshop

Andreas Tille

Short Introduction

Let's start with your own example

Communication

- **Debian Med user list**
- **Debian Med developer list**
- **Debian Med commits**
- #debian-med at irc.debian.org
- **Mentoring of Month (MoM)**
Debian Med packaging workshop

Andreas Tille

Short Introduction

Let's start with your own example

Communication

- Debian Med user list
- Debian Med developer list
- Debian Med commits
- #debian-med at irc.debian.org
- Mentoring of Month (MoM)
Communication

Debian Med user list
Debian Med developer list
Debian Med commits

#debian-med at irc.debian.org
Mentoring of Month (MoM)
Communication

- Debian Med user list
- Debian Med developer list
- Debian Med commits
- #debian-med at irc.debian.org
- Mentoring of Month (MoM)
Debian Med
packaging
workshop

Andreas Tille

Short
Introduction

Lets start with your own example

Communication

- Debian Med user list
- Debian Med developer list
- Debian Med commits
- #debian-med at irc.debian.org
- Mentoring of Month (MoM)
Debian Med

Debian packaging workshop

Andreas Tille

Short Introduction

Let's start with your own example

---

**dpkg-dev** needed tools

- `make` `debian/rules` is a Makefile
- `dh-make` create a template
- `debhelper` just use it
- `devscripts` very useful
- `fakeroot` pretend to be root
- `gnupg` sign package
- `lintian` policy checker
- `pbuilder` build package in chroot
- `build-essential` metapackage
Needed packages

```
dpkg-dev  needed tools

make  debian/rules is a Makefile

dh-make  create a template

debhelper  just use it

devscripts  very useful

fakeroot  pretend to be root

gnupg  sign package

lintian  policy checker

pbuilder  build package in chroot

build-essential  metapackage
```
Needed packages

*dpkg-dev* needed tools

`make` *debian/rules* is a Makefile

`dh-make` create a template

`debhelper` just use it

`devscripts` very useful

`fakeroot` pretend to be root

`gnupg` sign package

`lintian` policy checker

`pbuilder` build package in chroot

`build-essential` metapackage
Needed packages

- `dpkg-dev` needed tools
- `make` `debian/rules` is a Makefile
- `dh-make` create a template
- `debhelper` just use it
- `devscripts` very useful
- `fakeroot` pretend to be root
- `gnupg` sign package
- `lintian` policy checker
- `pbuilder` build package in chroot
- `build-essential` metapackage
Needed packages

- `dpkg-dev` needed tools
- `make` `debian/rules` is a Makefile
- `dh-make` create a template
- `debhelper` just use it
- `devscripts` very useful
- `fakeroot` pretend to be root
- `gnupg` sign package
- `lintian` policy checker
- `pbuilder` build package in chroot
- `build-essential` metapackage
Needed packages

\texttt{dpkg-dev} needed tools
\texttt{make} \texttt{debian/rules} is a \texttt{Makefile}
\texttt{dh-make} create a template
\texttt{debhelper} just use it
\texttt{devscripts} very useful
\texttt{fakeroot} pretend to be root
\texttt{gnupg} sign package
\texttt{lintian} policy checker
\texttt{pbuilder} build package in chroot
\texttt{build-essential} metapackage
Needed packages

- `dpkg-dev` needed tools
- `make` `debian/rules` is a Makefile
- `dh-make` create a template
- `debhelper` just use it
- `devscripts` very useful
- `fakeroot` pretend to be root
- `gnupg` sign package
- `lintian` policy checker
- `pbuilder` build package in chroot
- `build-essential` metapackage
## Needed packages

- **dpkg-dev** needed tools
- **make** `debian/rules` is a Makefile
- **dh-make** create a template
- **debhelper** just use it
- **devscripts** very useful
- **fakeroot** pretend to be root
- **gnupg** sign package
- **lintian** policy checker
- **pbuilder** build package in chroot
- **build-essential** metapackage
Needed packages

- `dpkg-dev` needed tools
- `make` `debian/rules` is a `Makefile`
- `dh-make` create a template
- `debhelper` just use it
- `devscripts` very useful
- `fakeroot` pretend to be root
- `gnupg` sign package
- `lintian` policy checker
- `pbuilder` build package in chroot
- `build-essential` metapackage
Needed packages

- dpkg-dev  needed tools
- make debian/rules is a Makefile
- dh-make create a template
- debhelper just use it
- devscripts very useful
- fakeroot pretend to be root
- gnupg sign package
- lintian policy checker
- pbuilder build package in chroot
- build-essential metapackage
What is your most wanted project?
How to start?

Let's start with your own example.

How to start?

dh_make

or

svn://svn.debian.org/svn/debian-med/trunk/package_template