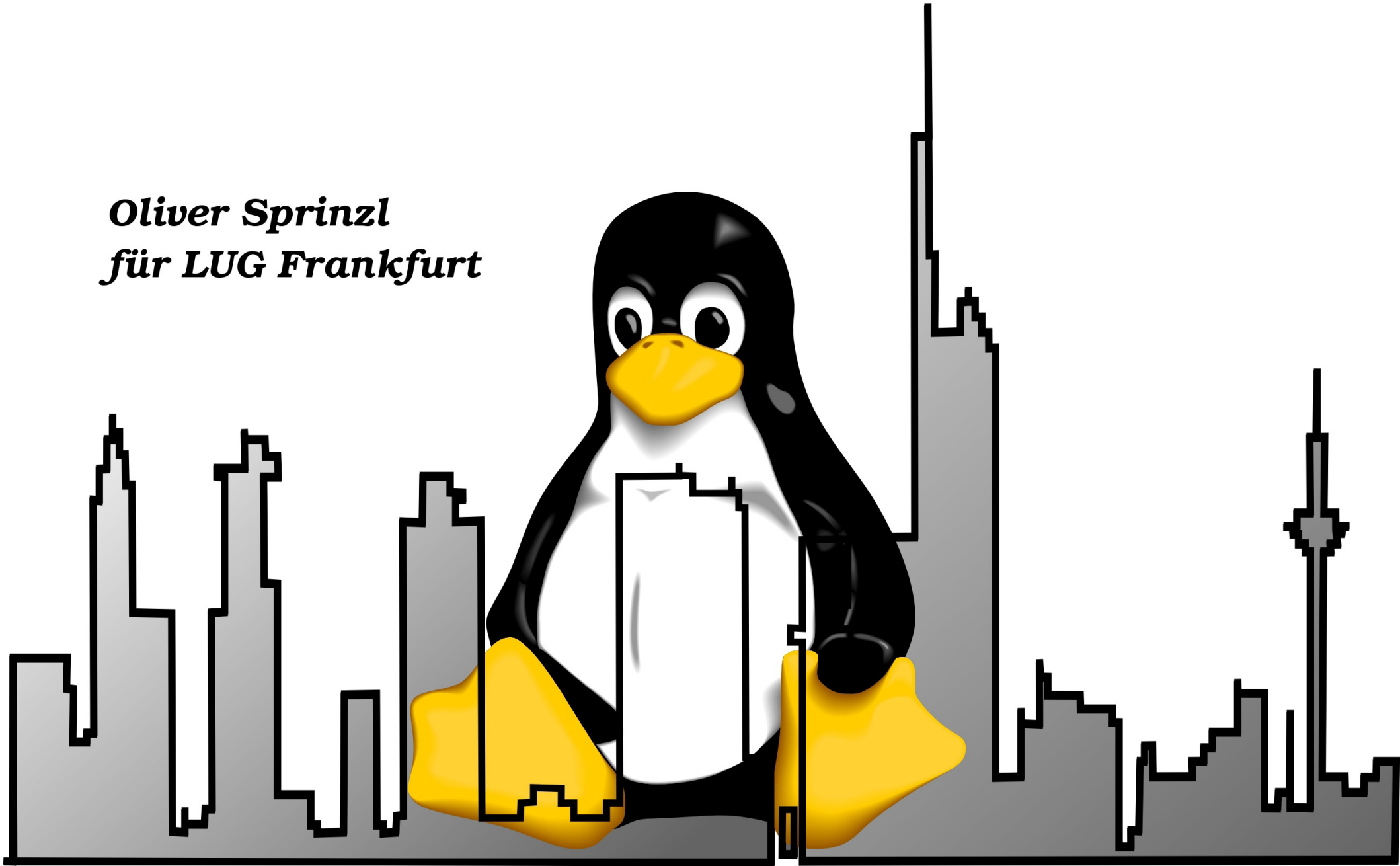
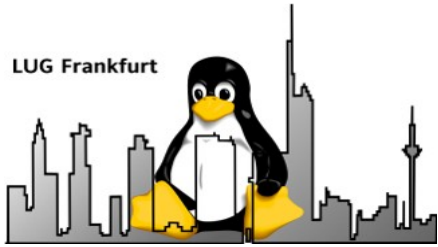


Hilfe, mein Programm gibt es nur im Source!  
Wir bauen Pakete ...

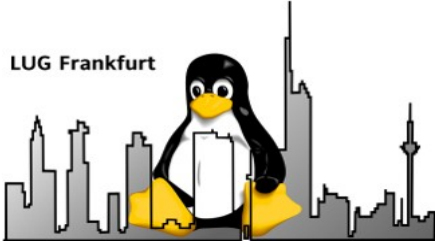
*Oliver Sprinzi  
für LUG Frankfurt*





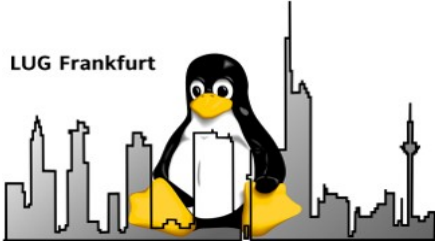
# Der Sinn von Paketen

- Sicherheit (signierte Pakete)
- Nachvollziehbarkeit (wo ist denn nun wieder?)
- Automatisierung (eigene Repositories)
- Auflösung der Paketabhängigkeiten



# Voraussetzungen

- Compiler und Buildtools (make & co.) müssen vorhanden sein:  
RPM: yum groupinstall development-tools  
DEB: apt-get install build-essentials
- ebenfalls diverse extra Tools:  
RPM: rpm-build, rpmddevtools, fedora-packager  
DEB: dh\_make, dpkg-dev, debhelper, devscripts
- bei Kernel Modulen: Kernel Source



# RPM Pakete bauen (1)

- Verzeichnisstruktur anlegen: Die Verzeichnisstruktur kann mit `rpmdev-setuptree` erstellt werden. Wenn als root aufgerufen, dann unter `/usr/src/redhat`, ansonsten im Home des aufrufenden Users. Struktur sieht so aus:

`$HOME/rpmbuild/BUILD`

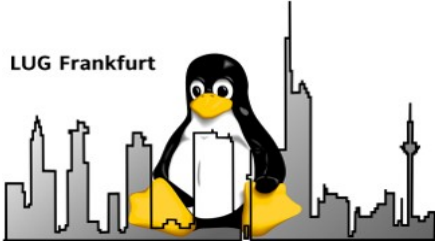
`RPMS`

`SOURCES`

`SPECS`

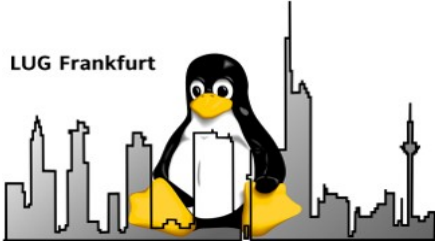
`SRPMS`

- RPM Defaults festlegen: `$HOME/.rpmmacros`



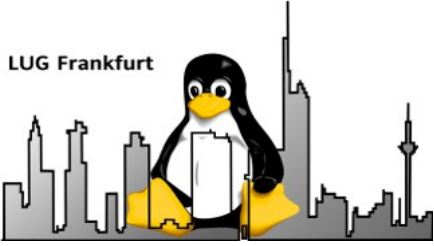
## RPM Pakete bauen (2)

- Source Code und Patches unter `$HOME/rpmbuild/SOURCES` oder `/usr/src/redhat/SOURCES` ablegen
- SPEC File in `$HOME/rpmbuild/SPECS` oder `/usr/src/redhat/SPECS` anlegen (Grundgerüst kann mit `rpmdev-newspec` erstellt werden)



## RPM Pakete bauen (3)

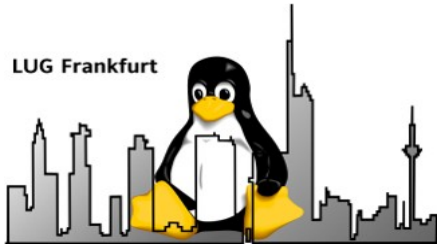
- SPEC File im Detail:  
SPEC Files werden in Sections aufgeteilt:  
HEADER mit %description, %prep, %build,  
%install, %files, %clean,  
%pre, %post, %preun, %postun



# RPM Pakete bauen (4)

## • Beispiel (Teil 1):

```
Name:                hello
Version:             2.4
Release:             1%{?dist}
Summary:             Hello world
Group:               Application/System
License:             GPLv3+
URL:                 http://savannah.gnu.org/projects/hello/
Source0:             %{name}-%{version}.tar.gz
# Patch0:            %{name}-info.patch
BuildRoot:           %{_tmppath}/%{name}-%{version}-%{release}-root-%(%{__id_u} -n)
BuildRequires:      autoconf >= 2.60
BuildRequires:      automake
BuildRequires:      gettext-devel >= 0.16
BuildRequires:      texinfo
# Requires:
%description
The hello command is for printing a greeting message.
```



# RPM Pakete bauen (5)

- **Beispiel Teil2:**

```
%prep
```

```
%setup -q
```

```
# %patch0 -p
```

```
%build
```

```
%configure
```

```
make %{?_smp_mflags}
```

```
%install
```

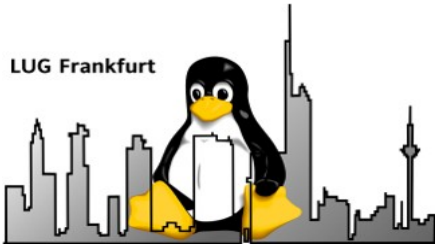
```
rm -rf $RPM_BUILD_ROOT
```

```
make install DESTDIR=$RPM_BUILD_ROOT
```

```
%clean
```

```
rm -rf $RPM_BUILD_ROOT
```





# RPM Pakete bauen (6)

- Beispiel Teil 3:

```
%files
```

```
%defattr(644,root,root,755)
```

```
%doc AUTHORS ChangeLog* NEWS README THANKS TODO
```

```
%attr(755,root,root) %{_bindir}/hello
```

```
%{_mandir}/man1/hello.1.gz
```

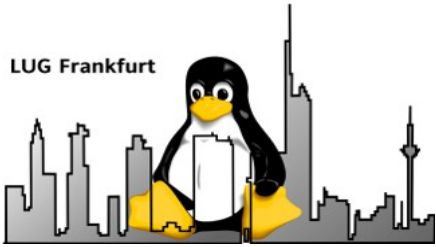
```
%{_infodir}/hello.info.gz
```

```
/usr/share/locale/
```

```
%changelog
```

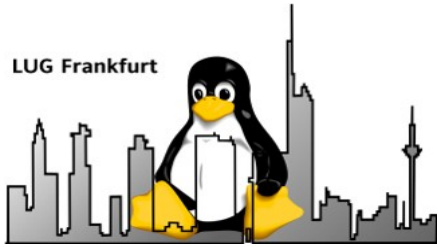
```
* Thu Dec 9 2008 Karl Berry <karl@gnu.org>
```

```
- Release Version 2.4.
```



## RPM Pakete bauen (7)

- Abschließend Paketgenerierung starten:  
`rpmbuild -ba <specfile>` (mit Source RPM)  
oder  
`rpmbuild -bb <specfile>` (ohne Source RPM)

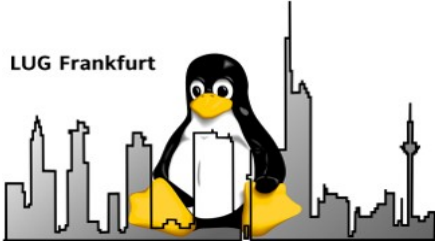


# DEB Pakete bauen (1)

- Zwei Möglichkeiten:

**dh\_make + debhelper**

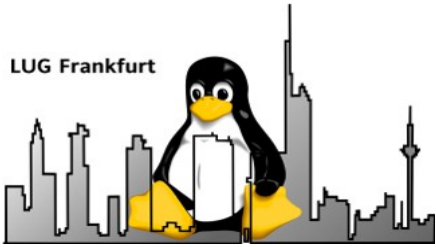
**checkinstall**



# DEB Pakete bauen (2)

## Möglichkeit 1 - dh\_make:

- Entpacken des Source Files, wichtig: entpacktes Verzeichnis muss `<Packagename>-<Version>` heißen
- `dh_make -e your.email@address -f ../package-0.0.1.tar.gz`
- Anpassen von control, copyright, changelog, rules



# DEB Pakete bauen (3)

- debian/control:

Source: hello

Section: base

Priority: optional

Maintainer: Richard Mustermann <richard@mustermann.de>

Build-Depends: debhelper (>= 7), autotools-dev

Standards-Version: 3.8.0

Homepage: <http://savannah.gnu.org/projects/hello/>

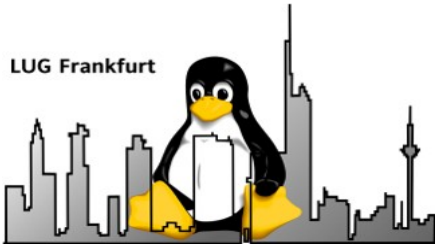
Package: hello

Architecture: any

Depends: \${shlibs:Depends}, \${misc:Depends}

Description: GNU Hello world

GNU Hello world



# DEB Pakete bauen (4)

- debian/copyright:

```
This package was debianized by Richard Mustermann  
<richard@mustermann.de> on
```

```
Mon, 19 Oct 2009 22:16:04 +0200.
```

```
It was downloaded from http://savannah.gnu.org/projects/hello/
```

```
Upstream Author(s):
```

```
    Karl Berry karl@gnu.org
```

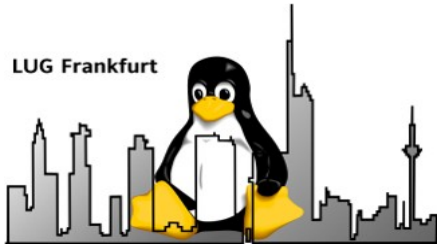
```
Copyright:
```

```
    Copyright © 2001-2009 Free Software Foundation, Inc.
```

```
License:
```

```
    hello is released under GPL Version 3+
```

```
The Debian packaging is copyright 2009, Richard Mustermann  
<richard@mustermann.de> and is licensed under the GPL, see  
' /usr/share/common-licenses/GPL'.
```



## DEB Pakete bauen (5)

- debian/changelog:

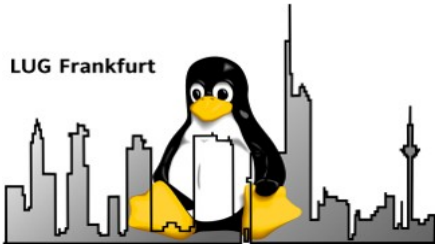
```
hello (2.4-1) unstable; urgency=low
```

```
* Initial release
```

```
* Fixed Bug #01234
```

```
-- Richard Mustermann <richard@mustermann.de>
```

```
Mon, 19 Oct 2009 22:16:04 +0200
```

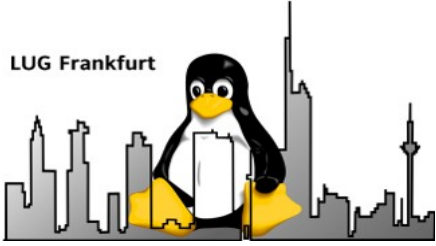


# DEB Pakete bauen (6)

- debian/rules:

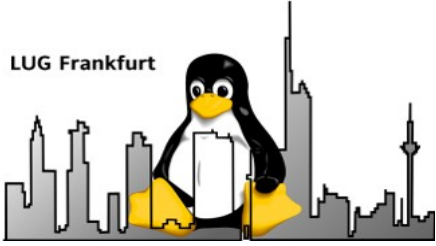
```
#!/usr/bin/make -f
DEB_HOST_GNU_TYPE    ?= $(shell dpkg-architecture -qDEB_HOST_GNU_TYPE)
DEB_BUILD_GNU_TYPE   ?= $(shell dpkg-architecture -qDEB_BUILD_GNU_TYPE)
ifneq ($(DEB_HOST_GNU_TYPE),$(DEB_BUILD_GNU_TYPE))
CROSS= --build $(DEB_BUILD_GNU_TYPE) --host $(DEB_HOST_GNU_TYPE)
else
CROSS= --build $(DEB_BUILD_GNU_TYPE)
endif
config.status: configure
dh_testdir
# Add here commands to configure the package.
ifneq "$(wildcard /usr/share/misc/config.sub)" ""
cp -f /usr/share/misc/config.sub config.sub
endif
ifneq "$(wildcard /usr/share/misc/config.guess)" ""
cp -f /usr/share/misc/config.guess config.guess
endif
./configure $(CROSS) --prefix=/usr --mandir=\${prefix}/share/man --infodir=\${prefix}/share/info
CFLAGS="$CFLAGS" LDFLAGS="-Wl,-z,defs"
```





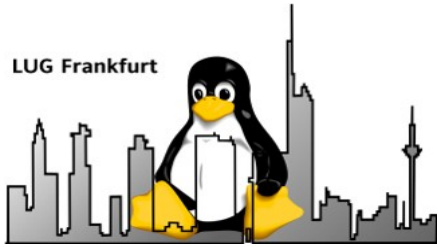
# DEB Pakete bauen (7)

```
build: build-stamp
build-stamp: config.status
dh_testdir
# Add here commands to compile the package.
$(MAKE)
#docbook-to-man debian/hello.sgml > hello.1
touch $@
clean:
dh_testdir
dh_testroot
rm -f build-stamp
# Add here commands to clean up after the build process.
[ ! -f Makefile ] || $(MAKE) distclean
rm -f config.sub config.guess
dh_clean
```



# DEB Pakete bauen (8)

```
install: build
dh_testdir
dh_testroot
dh_prep
dh_installdirs
# Add here commands to install the package into debian/hello.
$(MAKE) DESTDIR=$(CURDIR)/debian/hello install
# Build architecture-independent files here.
binary-indep: install
# We have nothing to do by default.
# Build architecture-dependent files here.
binary-arch: install
dh_testdir
dh_testroot
dh_installchangelogs ChangeLog
dh_installdocs
dh_installexamples
```



# DEB Pakete bauen (9)

`dh_installman`

`dh_link`

`dh_strip`

`dh_compress`

`dh_fixperms`

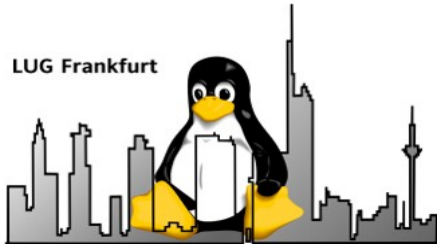
`dh_installdeb`

`dh_shlibdeps`

`dh_gencontrol`

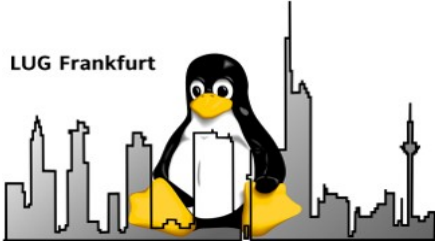
`dh_md5sums`

`dh_builddeb`



## DEB Pakete bauen (10)

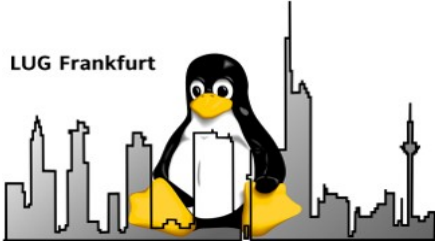
- Abschließend Paketgenerierung starten:  
`dpkg-buildpackage -rfakeroot`  
bzw.  
`fakeroot debian/rules binary`  
(erstellt nur das .deb File)



# DEB Pakete bauen (11)

## Möglichkeit 2 - checkinstall:

- Beim `./configure`, `make` und `make install`, das `make install` weglassen und statt dessen `checkinstall` aufrufen



# DEB Pakete bauen (12)

## Checkinstall Menü:

checkinstall 1.6.1, Copyright 2002 Felipe Eduardo Sanchez Diaz Duran

This package will be built according to these values:

0 - Maintainer: [ richard@mustermann.de ]

1 - Summary: [ hello world ]

2 - Name: [ hello ]

3 - Version: [ 2.4 ]

4 - Release: [ 1 ]

5 - License: [ GPL ]

6 - Group: [ base ]

7 - Architecture: [ amd64 ]

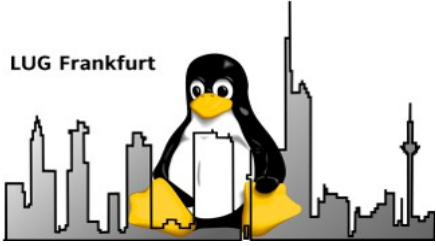
8 - Source location: [ hello-2.4 ]

9 - Alternate source location: [ ]

10 - Requires: [ ]

11 - Provides: [ hello ]

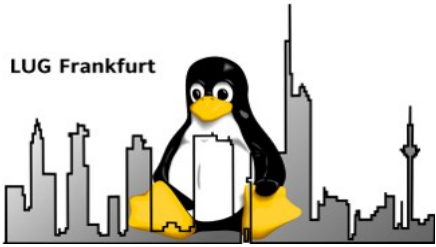
Enter a number to change any of them or press ENTER to continue:



# Pakete überprüfen

- DEB Pakete überprüfen: lintian
- RPM Pakete überprüfen: rpmlint

dann: Probleme korrigieren ...



# Links

- <https://wiki.ubuntu.com/PackagingGuide/Complete>
- [http://wiki.ubuntuusers.de/Programme\\_kompilieren](http://wiki.ubuntuusers.de/Programme_kompilieren)
- <http://www.debian.org/doc/maint-guide/>
- <https://fedoraproject.org/wiki/Packaging/Guidelines>
- <https://fedoraproject.org/wiki/PackageMaintainers/CreatingPackageHowTo>
- <http://www.rpm.org/max-rpm/>
- <http://www.ibm.com/developerworks/linux/library/l-rpm1/>
- <http://www.ibm.com/developerworks/linux/library/l-rpm2/>
- <http://www.ibm.com/developerworks/linux/library/l-rpm3/>
- <http://www.pathname.com/fhs/>



Fragen ???

Danke  
fürs  
Mitnehmen  
und  
Tschüß

LUG Frankfurt

