

# Einführung in Ubuntu

| REVISION HISTORY |            |             |             |
|------------------|------------|-------------|-------------|
| NUMBER           | DATE       | DESCRIPTION | NAME        |
| 0.1              | 2013-08-02 |             | Jörg Reuter |

## Contents

|          |  |          |
|----------|--|----------|
| <b>1</b> | <b>3. Unterrichtseinheit</b>             | <b>1</b> |
| 1.1      | Softwareverwaltung . . . . .             | 1        |
| 1.2      | Komponenten . . . . .                    | 2        |
| 1.3      | Paketquellen . . . . .                   | 2        |
| 1.4      | PPA . . . . .                            | 3        |
| 1.5      | sources.list . . . . .                   | 3        |
| 1.6      | Metapakete . . . . .                     | 3        |
| 1.7      | Updates . . . . .                        | 3        |
| 1.8      | Pakete suchen und installieren . . . . . | 3        |
| 1.9      | Lokale Paketquellen . . . . .            | 4        |

---



Jörg Reuter ist seit 2000 Studienrat an der Ferdinand-Braun-Schule in Fulda und unterrichtet dort Anwendungsentwicklung (C und Java) und Netzwerktechnik (Linux), Elektrotechnik (Berufsschule), Politik und katholische Religion (Fachoberschule).

Geboren 1970 in Gießen, 1990/1991 Zivildienst (24 Monate) am Universitätsklinikum Gießen in der Hals- Nasen- und Ohrenklinik. Während des Zivildienstes Fernstudium an der Fernuniversität Hagen (Elektrotechnik). 1991-1997 Studium der Elektrotechnik an der TU Darmstadt. 1997/1998 für einen privaten Bildungsträger in Gießen tätig. 1998 Referendar an der Ferdinand-Braun-Schule.

#### Dieses Dokument

Dieses Dokument wurde in AsciiDoc erstellt und seine aktuelle Fassung ist unter [Github](#) verfügbar.

## 1 3. Unterrichtseinheit

Image virtuelle Maschine: <http://labs.fbs-fulda.info/Images> Benutzername/Passwort: fbs

Youtube-Kanal: [http://www.youtube.com/channel/UCd1\\_XM1QkdeHbZAy7UzN2Ew?feature=watch](http://www.youtube.com/channel/UCd1_XM1QkdeHbZAy7UzN2Ew?feature=watch)

### 1.1 Softwareverwaltung

Unter Ubuntu wird Software über Paketquellen zur Verfügung gestellt. paketquellen sind das Medium auf dem sich die Software befindet wie Internetadressen, DVD oder USB-Sticks etc.

Die Software an sich wird in Archive unterteilt die wiederum in Komponenten unterteilt sind. Ein Eintrag in der Konfigurationsdatei (/etc/apt/sources.list) sieht wie folgt aus

deb <http://forsthaus-bimbach.de/ubuntu/> precise-security main restricted universe multiverse

Am Anfang der Zeile steht die Quelle, hier eine Internetadresse, danach das Archiv (hier precise-security) und danach die entsprechende Komponente (hier main restricted universe multiverse). Rein praktisch sieht die Dateistruktur wie auf dem Bild aus: Unter der Version kommen die Archive, darunter die Komponenten, dann die Computer-Architektur (ARM, i386 oder amd64 etc.). Nach der Architektur kommt eine Datei die alle Pakete mit entsprechenden Verweise in das Verzeichnis pool enthält. Im Verzeichnis pool sind alle Pakete nach Komponenten und dann alphabetisch unterteilt. Dann kommen die Verzeichnisse der einzelnen Programme.

```
Links      Datei      Befehl      Optionen      Rechts
<- /var/www/ubuntu/pool/main/e/e2fsprogs .[^]>

'n          Name          Größe  Modifikations
/..         ÜBERVZ. 29. Jul 12:00
/jabber.py  4096    29. Jul 11:59
/jack-audio-connection-kit 4096    29. Jul 11:59
/jackd2     4096    29. Jul 11:59
/jade       4096    29. Jul 11:59
/jadetex    4096    17. Aug 23:19
/jargs      4096    29. Jul 11:59
/jarjar     4096    29. Jul 11:59
/jasper     4096    29. Jul 11:59
/jasper-initramfs 4096    29. Jul 11:59
/java-atk-wrapper 4096    29. Jul 11:59
/java-common 4096    29. Jul 11:59
/java-wrappers 4096    29. Jul 11:59
/javacc     4096    29. Jul 11:59
/javatools  4096    29. Jul 11:59
/jaxme      4096    29. Jul 11:59
/jbig2dec   4096    29. Jul 11:59
/jflex      4096    29. Jul 11:59
/jfsutils   4096    29. Jul 11:59
/jiglit     4096    29. Jul 11:59
/jinja2     4096    29. Jul 11:59
/jllex      4096    29. Jul 11:59
/jline      4096    29. Jul 11:59
/jmock      4096    29. Jul 11:59
/jockey     4096    22. Sep 19:44
/john       4096    29. Jul 11:59
/jovie      4096    29. Jul 11:59
/joystick   4096    29. Jul 11:59
/jquery     4096    29. Jul 11:59
/jsch       4096    29. Jul 11:59
/json-c     4096    29. Jul 11:59
/json-glib  4096    29. Jul 11:59

ÜBERVZ.

645G/902G (71%)

Verzeichnisbaum
└─ ubuntu
   └─ dists
      ├── precise
      ├── precise-backports
      ├── precise-proposed
      ├── precise-security
      ├── main
      │   ├── binary-amd64
      │   └── binary-i386
      ├── multiverse
      ├── restricted
      ├── universe
      └── precise-updates
   └─ pool
      └─ main
         ├── a
         ├── b
         ├── c
         ├── d
         ├── e
         ├── e2fsprogs
         ├── easymock
         ├── ebook-tools
         ├── ebttables
         ├── ecj
         ├── ecryptfs-utils
         ├── ed
         ├── edubuntu-live
         ├── efibootmgr
         ├── egenix-mx-base
         ├── egllibc
         ├── eigen2
         └── eject

/var/www/ubuntu/pool/main/e/e2fsprogs

Hint: You can browse RPM files by tapping enter on top of an rpm file.
root@forsthaus-bimbach: /var/www/ubuntu/pool/main/e/e2fsprogs#
```

- precise: Die Original-Quelle, die nach veröffentlichung einer Ubuntu-Version nicht mehr verändert wird.
- precise-security: Aktualisierungen, die zwingend Notwendig sind.
- precise-updates: Fehlerbereinigte Pakete. Normalerweise keine neue Versionen.
- precise-backports: Neuere Versionen von Programmen.
- precise-proposed: Nicht verwenden! Es dient zum testen von Software.

## 1.2 Komponenten

Alle Pakete sind vier Komponenten zugeordnet, die bei allen Distributionen bisher gleich sind. Sie sind gegliedert nach dem Grad des Supports der Entwickler von Ubuntu und Lizenzanforderungen:

- main: Entsprechen den Lizenzanforderungen von Ubuntu und werden von den Entwicklern unterstützt
- restricted: Erfüllen nicht die Lizenzanforderungen, werden aber von den Entwicklern wegen ihrer Bedeutung unterstützt.
- universe: Freie Software, die nicht direkt von den Entwicklern von ubuntu unterstützt wird.
- multiverse: Unfreie Software die nicht von den Entwicklern unterstützt wird. bekannt ist die Quelle vor allem wegen Multi-mediacodecs
- partner: Kommerzielle, kostenpflichtige Software.

### 1.3 Paketquellen

Siehe [http://wiki.ubuntuusers.de/Paketquellen\\_freischalten#Terminal](http://wiki.ubuntuusers.de/Paketquellen_freischalten#Terminal)

Aufbau des Befehls:

```
sudo add-apt-repository deb PFAD/ZUR/QUELLE DISTRIBUTION KOMPONENTEN
```

Beispiel:

```
sudo add-apt-repository deb http://archive.ubuntu.com/ubuntu precise main restricted universe multiverse
```

Oft ist es nötig, einen Key einzufügen. Dies geschieht mit dem Befehl:

```
sudo apt-key adv --keyserver PGP_KEY_SERVER --recv-keys ID
```

Beispiel:

```
sudo apt-key adv --recv-keys --keyserver keyserver.ubuntu.com A1715D88E1DF1F24
```

## 1.4 PPA

(siehe auch [http://wiki.ubuntuusers.de/Paketquellen\\_freischalten/PPA](http://wiki.ubuntuusers.de/Paketquellen_freischalten/PPA))

PPA (Personal Package Archive) ermöglicht Entwicklern Quellpakete auf den Server zu laden und das System baut für verschiedene Architekturen und Versionen die entsprechenden Installationspakete.

Der Einsatz von PPA-Paketen ist bei Produktivsystemen verlangt erhöhte Vorsicht, da die Überprüfung der Pakete auf Sicherheitslücken etc. nicht immer gewährleistet ist.

Hinzufügen entsprechender Quellen:

```
sudo add-apt-repository ppa:LP-BENUTZER/PPA-NAME
```

## 1.5 sources.list

In der Datei sources.list im Verzeichnis /etc/apt und den Unterverzeichnissen sind alle Paketquellen verzeichnet und können bei Bedarf auch im Editor abgeändert werden. Siehe auch <http://wiki.ubuntuusers.de/sources.list>

## 1.6 Metapakete

Metapakete beinhalten sehr viele Programme. Z.B. das Metapaket ubuntu-desktopm enthält die komplette graphische Oberfläche mit allen Programmen die nötig sind wie Thunderbird, firefox etc. Metapakete ersparen also die Installation vieler kleiner Pakete und ermöglichen es mit einem kurzen Befehl sehr viele häufig gebrauchte Programmpakete zu installieren. Mehr dazu: <http://wiki.ubuntuusers.de/Metapakete>

## 1.7 Updates

```
apt-get update apt-get upgrades
```

Manche Pakete wie ein neuer Kernel werden erst nach aufruf des Befehls:

```
apt-get dist-upgrade
```

installiert.

## 1.8 Pakete suchen und installieren

Mit dem Befehl

```
apt-get install apache 2
```

wird der Webserver Apache2 installiert.

Das Suchen nach Paketen:

```
apt-cache search apache
```

werden alle Pakete gesucht, bei denen der Name apache auftaucht.

## 1.9 Lokale Paketquellen

Da es immer wieder Probleme mit der Internetverbindung in der Schule gibt, wird dringend empfohlen bei Referaten etc lokale paketquellen zu verwenden. mehr dazu findet Ihr hier: [http://wiki.ubuntuusers.de/Lokale\\_Paketquellen](http://wiki.ubuntuusers.de/Lokale_Paketquellen)

== Copyright

Das Copyright an diesem Skript liegt bei Jörg Reuter, Ferdinand-Braun-Schule Fulda.

Dieses Skript ist veröffentlicht unter der creative commons Lizenz  
*Namensnennung - Weitergabe unter gleichen Bedingungen* (by-sa).

CC by sa

Eine Beschreibung der Lizenz finden Sie hier:

<http://creativecommons.org/licenses/by-sa/3.0/de/>

Der rechtsverbindliche Text der Lizenz ist hier:

<http://creativecommons.org/licenses/by-sa/3.0/de/legalcode>

Die Software von Jörg Reuter ist in weiten Teilen Open Source.

Wenn nicht anders angegeben, ist der Quellcode veröffentlicht unter der GPLv3:

Der rechtsverbindliche Text der GPLv3 Lizenz ist hier:

<http://www.gnu.org/licenses/gpl.html>

Deutsche Übersetzung:

<http://www.gnu.de/documents/gpl.de.html>

---