

# Installationsanleitung

## Webanwendung zur Extraktion von Teildatensätzen aus DBpedia

Robert Bielinski    Marvin Hofer    Till Nestler    Hans Angermann  
Christian Ernst    Dominik Strohscheer    Jonas Rebmann

## Inhaltsverzeichnis

<b>Bemerkung</b>	<b>1</b>
<b>System</b>	<b>1</b>
<b>Abhängige Pakete installieren</b>	<b>2</b>
<b>Vorbereitung für Apache Railsapp deployment</b>	<b>3</b>
<b>Apache und Passenger konfigurieren</b>	<b>4</b>

## Bemerkung

Installation des im Project wetd16 erstellten Railsframeworks unter einem Linux Debian 8 oder Ubuntu 15.04 Server.

## System

Ubuntu 15.04 / Debian 8

Aktualisieren des Systems:

```
1 $ sudo apt-get update
2 $ sudo apt-get upgrade
```

Nutzer für Framework erstellen, mit Root Rechten

```
1 $ sudo adduser railsuser
2 $ sudo adduser railsuser sudo
```

In den Benutzer und sein Hauptverzeichnis wechseln

```
1 $ su railsuser
2 $ cd ~
```

## Abhängige Pakete installieren

```
1 $ sudo apt-get install git-core curl zlib1g-dev build-essential libssl-dev
   libreadline-dev libyaml-dev libsqlite3-dev sqlite3 libxml2-dev
   libxslt1-dev libcurl4-openssl-dev python-software-properties libffi-dev
```

## Ruby und Rails installieren

```
1 $ sudo apt-get install ruby ruby-dev
2 $ sudo gem install rails
```

Rails ohne RI mit

```
1 $ sudo gem install rails --no-ri
```

## Node.js installieren

Installation von node.js für Rails

```
1 $ sudo add-apt-repository ppa:chris-lea/node.js
2 $ sudo apt-get update
3 $ sudo apt-get install nodejs
```

## Java installieren

Java Version > 1.8 für Maven / RDFUnit

```
1 $ sudo echo "deb http://ppa.launchpad.net/webupd8team/java/ubuntu xenial
   main" | tee /etc/apt/sources.list.d/webupd8team-java.list
2 $ sudo echo "deb-src http://ppa.launchpad.net/webupd8team/java/ubuntu
   xenial main" | tee -a /etc/apt/sources.list.d/webupd8team-java.list
3 $ sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys
   EEA14886
4 $ sudo apt-get update
5 $ sudo apt-get install oracle-java8-installer
```

Java Version checken

```
1 $ java -version
```

## Maven für RDFUnit installieren

Für RDFUnit

```
1 $ sudo apt-get install maven
```

## Framework laden

Git clonen

```
1 $ cd ~  
2 $ git clone wetd16@pcai042.informatik.uni-leipzig.de:wetd16code.git
```

mit yes bestätigen, Passwort nötig!

```
1 $ cd wetd16code/railsapp  
2 $ sudo bundle install
```

## RDFUnit compilieren

RDF-Unit-0.8 im wetd16code/railsapp/app/models/RDFUnit-0.8/Ordner

```
1 $ cd app/models/RDFUnit-0.8/  
2 $ sudo mvn clean install
```

## Testen der Railsapp

```
1 $ cd ~/wetd16code/railsapp/  
2 $ sudo bundle exec passenger start
```

localhost:3000 aufrufen oder

```
1 $ curl localhost:3000
```

## Vorbereitung für Apache Railsapp deployment

```
1 $ rake secret
```

diesen key kopieren config/secrets.yml öffnen und KEY einfügen

```
1 $ nano config/secrets.yml
```

Zeile:

```
production: secret_key_base: "HIER KEY EINFÜGEN"
```

nano mit [STRG+O] speichern und mit [STRG+X] verlassen

Assets vorcompilieren.

```
1 $ RAILS_ENV=production bundle exec rake assets:precompile
```

wenn nicht möglich dann als Root.

## Apache und Passenger konfigurieren

```
1 $ sudo apt-get install apache2 apache2-dev
```

Passenger Global installieren

```
1 $ sudo gem install passenger
```

## Passenger Modul für Apache installieren

```
1 $ sudo passenger-install-apache2-module
```

Der Anleitung folgen

Python, Node.js, Rails auswählen.

Wichtig: Einfügen des "LoadModule passenger\_module" Teils in /etc/apache2/apache2.conf nicht vergessen. Aufforderung dazu kommt während passenger-install-apache2-module

## Apache2.conf anpassen

```
1 $ cd /etc/apache2/
```

wenn keine Berechtigung für /etc/apache2/ dann als Root

```
1 $ sudo nano apache2.conf
```

apache2.conf bearbeiten und unten anfügen:

```
1 <VirtualHost *:80>
2     ServerName S_NAME
3     DocumentRoot PATH
4     <Directory PATH >
5         AllowOverride all
6         Options MultiViews
7         Order allow,deny
```

```
8         Allow from all
9         Require all granted
10     </Directory>
11 </VirtualHost>
```

PATH ist Pfad zu rails public Ordner. Beispielpfad = /home/railsuser/wetd16code/railsapp/public

S\_NAME ist Name unter dem der Server erreichbar sein soll. Beispielname = example.com oder localhost

## Rechte des Rails Ordners an Apache übergeben:

```
1 $ sudo chown -R www-data:www-data /home/railsuser/wetd16code/
```

## Apache2 Neustarten:

```
1 $ sudo apachectl restart
```

oder

```
1 $ sudo apachectl graceful
```