



USING ROCKER WITH PPAs FOR FUN AND PROFIT

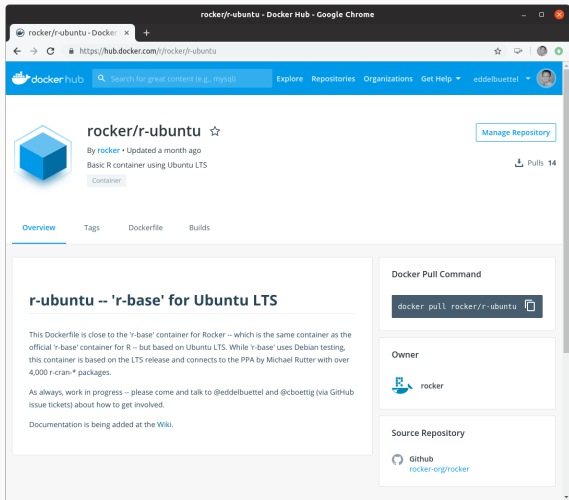
INSTALLING R PACKAGES NOW SIMPLE, FAST, EASY AND RELIABLE

Dirk Edelbuettel

R⁴ Video Lightning Talk

June 9, 2019

USES BASIC ROCKER CONTAINER FOR R AND UBUNTU



The screenshot shows the Docker Hub page for the repository `rocker/r-ubuntu`. The page title is "rocker/r-ubuntu - Docker Hub - Google Chrome". The URL is `https://hub.docker.com/r/rocker/r-ubuntu`. The repository is owned by `rocker` and is updated a month ago. It is described as a "Basic R container using Ubuntu LTS". The page includes a "Manage Repository" button, a "Pulls 14" indicator, and navigation tabs for "Overview", "Tags", "Dockerfile", and "Builds". The main content area features the title "r-ubuntu -- 'r-base' for Ubuntu LTS" and a description: "This Dockerfile is close to the 'r-base' container for Rocker -- which is the same container as the official 'r-base' container for R -- but based on Ubuntu LTS. While 'r-base' uses Debian testing, this container is based on the LTS release and connects to the PPA by Michael Rutter with over 4,000 r-cran.* packages. As always, work in progress -- please come and talk to @eddelbuettel and @cboettig (via GitHub issue tickets) about how to get involved. Documentation is being added at the Wiki." To the right, there is a "Docker Pull Command" section with the command `docker pull rocker/r-ubuntu`, an "Owner" section listing `rocker`, and a "Source Repository" section pointing to `rocker-org/rocker` on GitHub.

Rocker Container

Rocker container
based on Ubuntu
18.04 LTS

Added 'marutter'
PPA information

More details are on
the next few slides

An Introduction to Rocker: Docker Containers for R

by Carl Boettiger, Dirk Eddelbuettel

Abstract We describe the Rocker project, which provides a widely-used suite of Docker images with customized R environments for particular tasks. We discuss how this suite is organized, and how these tools can increase portability, scaling, reproducibility, and convenience of R users and developers.

Introduction

The Rocker project was launched in October 2014 as a collaboration between the authors to provide high-quality Docker images containing the R environment (Boettiger and Eddelbuettel, 2014). Since that time, the project has seen both considerable uptake in the community and substantial development and evolution. Here we seek to document the project's objectives and uses.

What is Docker?

Docker is a popular open-source tool to create, distribute, deploy, and run software applications using containers. Containers provide a virtual environment (see Clark et al. (2014) for an overview of common virtual environments) requiring all operating-system components an application needs to run. Docker containers are lightweight as they share the operating system kernel, start instantly using a layered filesystem which minimizes disk footprint and download time, are built on open standards that run on all major platforms (Linux, Mac, Windows), and provide an added layer of security by running an application in an isolated environment (Docker, 2015). Familiarity with a few key terms is helpful in understanding this paper. The term "container" refers to an isolated software environment on a computer. R users can think of running a container as analogous to loading an R package; a container is an active instance of a static Docker image. A Docker "image" is a binary archive of that software, analogous to an R binary package; a given version is downloaded only once, and can then be "run" to create a container whenever it is needed. A "Dockerfile" is a recipe, the source-code, to create a Docker image. Pre-built Docker images are publicly available through Docker Hub, which plays a role for central distribution similar to CRAN in our analogy. Development and contributions to the Rocker project focus on the construction, organization and maintenance of these Dockerfiles.

Rocker: Docker for R

See Boettiger and Eddelbuettel (2017) for a succinct introduction to Rocker.

The container used here is 'base layer': standard Ubuntu 18.04 plus the 'marutter' PPAs referenced in the Ubuntu README at CRAN

The screenshot shows the Launchpad page for the PPA 'cran2deb4ubuntu_3.5' owned by Michael Rutter. The page includes a description, instructions on how to add the PPA to the system, activity statistics, and an overview of published packages.

PPA description
8 packages for Ubuntu LTS. Based on OSM Task Views.

Adding this PPA to your system
You can update your system with unsupported packages from this untrusted PPA by adding `ppa:marutter/c2d4u.5` to your system's Software Sources. [Please read about installing](#)

```
sudo add-apt-repository ppa:marutter/c2d4u.5
sudo apt-get update
```

PPA statistics
Activity
1158 updates added during the past month.

Latest updates

Package	Version	Uploaded by
libEyr	0.5.4-3cran1ppa@bionic	Michael Rutter (2018-05-31)
abc	2.3-3cran1ppa@bionic	Michael Rutter (2018-05-16)
abc-data	1.0-3cran1ppa@bionic	Michael Rutter (2018-05-16)

marutter c2d4u PPA

Using Ubuntu 18.04, we can deploy the `marutter c2d4u` PPA.

It provides over 4,000 `r-cran-*` binary packages with proper dependencies automatically resolved within Ubuntu 18.04.

Once cumbersome and error-prone installation becomes trivial

The live demo shows

```
apt-get install -y r-cran-tidyverse  
apt-get install -y r-cran-rstan
```

The first is a 50+ mb download with 200+ mb deployed, it installs in just over two minutes on my machines. It could no be easier.

NB: As an aside, we still think that *in general* fewer dependencies imply fewer possible breaking points – but this examples makes for a good demo.

Key Benefits

- We can create Docker containers for reproducible workloads
 - either interactively and then saving the Docker instance
 - or programmatically both locally or via Github / Docker hub
- These Docker containers can run wherever Docker runs:
 - Any OS: Linux or Windows or macOS hosts
 - Cloud deployment, Travis CI, Gitlab runners, ...
 - Easily shared with colleagues, students, reviewers, ...
 - You name it!

Complete set of commands used

```
## launch r-ubuntu:18.04 container for interactive use
docker run --rm -ti rocker/r-ubuntu:18.04 bash
## update inside container, update to current repo state
apt-get update; apt-get dist-upgrade -y
## install tidyverse binary package, ditto for rstan
apt-get install -y r-cran-tidyverse
apt-get install -y r-cran-rstan
```

THANK YOU!

slides <http://dirk.eddelbuettel.com/presentations/>

web <http://dirk.eddelbuettel.com/>

mail dirk@eddelbuettel.com

github [@eddelbuettel](#)

twitter [@eddelbuettel](#)