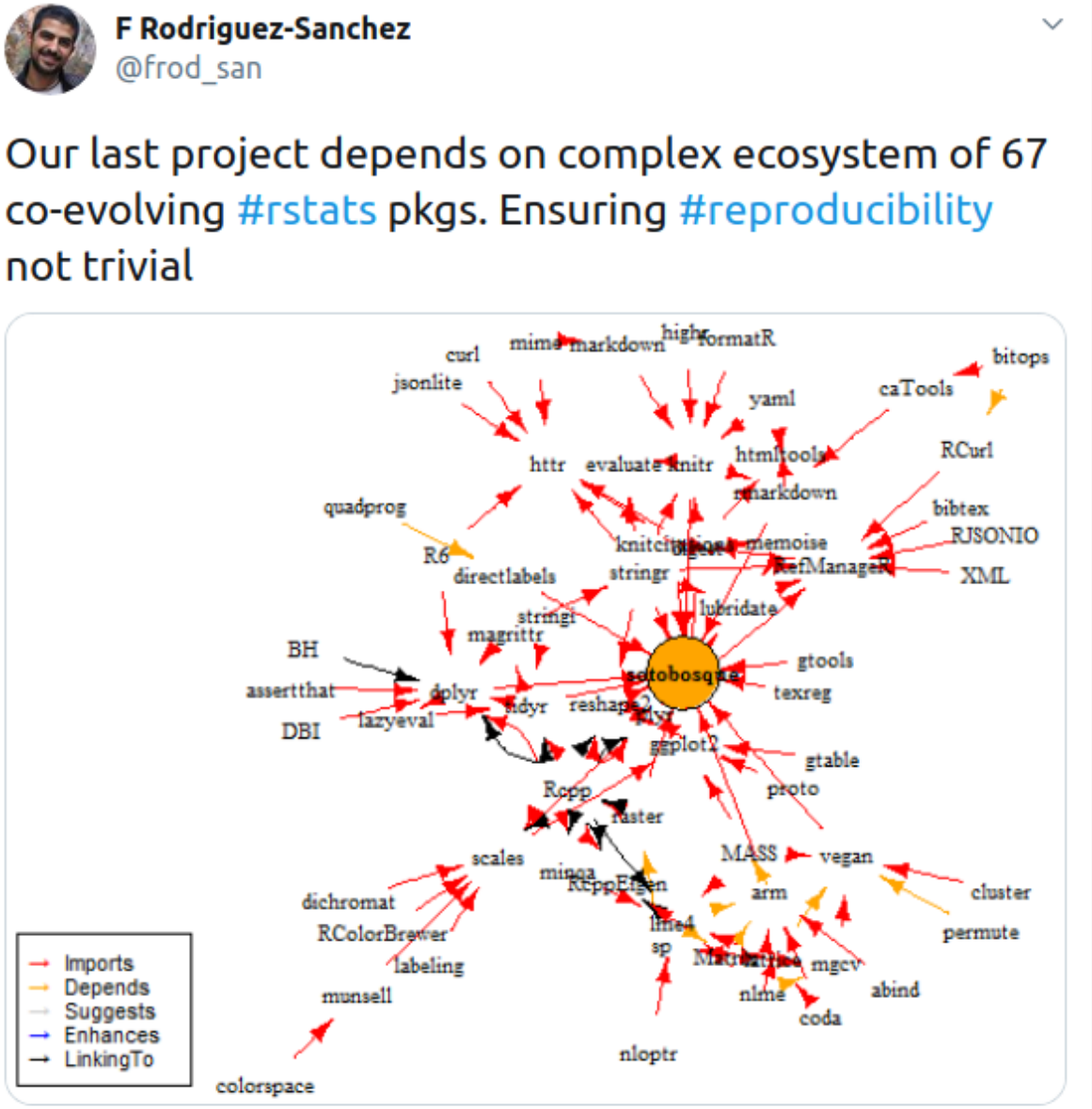
Controlling dependencies

Francisco Rodriguez-Sanchez

https://frodriguezsanchez.net



F Rodriguez-Sanchez
@frod_san

Our last project depends on complex ecosystem of 67 co-evolving #rstats pkgs. Ensuring #reproducibility not trivial

5:39 pm · 27 Jan 2016 · Twitter Web Client

Updating R packages broke your script?

Need to run an old script from you, or someone else?

How to reproduce your analysis in a year,
or different computer?

`sessionInfo` records OS & used packages

```
sessionInfo()
```

```
R version 4.4.2 (2024-10-31)
Platform: x86_64-pc-linux-gnu
Running under: Ubuntu 20.04.6 LTS

Matrix products: default
BLAS:   /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/liblapack.so.3;  LAPACK version 3.9.0

locale:
 [1] LC_CTYPE=en_GB.UTF-8       LC_NUMERIC=C
 [3] LC_TIME=es_ES.UTF-8        LC_COLLATE=en_GB.UTF-8
 [5] LC_MONETARY=es_ES.UTF-8    LC_MESSAGES=en_GB.UTF-8
 [7] LC_PAPER=es_ES.UTF-8       LC_NAME=C
 [9] LC_ADDRESS=C               LC_TELEPHONE=C
[11] LC_MEASUREMENT=es_ES.UTF-8 LC_IDENTIFICATION=C

time zone: Europe/Madrid
tzcode source: system (glibc)

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] knitr_1.49

loaded via a namespace (and not attached):
 [1] compiler_4.4.2    fastmap_1.2.0     cli_3.6.3         htmltools_0.5.8.1
 [5] tools_4.4.2       rstudioapi_0.17.1 yaml_2.3.10       codetools_0.2-20
 [9] rmarkdown_2.29    binb_0.0.7        xfun_0.50         digest_0.6.37
[13] rlang_1.1.4       evaluate_1.0.1
```

`checkpoint` recreates R packages in given date

```
library('checkpoint')

options(checkpoint.mranUrl="https://packagemanager.posit.co/")

checkpoint('2024-10-08')

source('analysis.R')
```

1. Detects packages used
2. Installs version from given date (only CRAN)
3. Independent install (not messing w/ main library)

automagic records & install packages (CRAN + GitHub)

```
automagic::make_deps_file()
```

File `deps.yaml` records dependencies:

```
- Package: equatiomatic
  Repository: CRAN
  Version: 0.1.0

- Package: report
  GithubUsername: easystats
  GithubRepo: report
  GithubRef: HEAD
  GithubSHA1: c48a4bb0a40df7116bc502aa3ce2cbbc9d70b7e2
```

To install all those dependencies:

```
automagic()
```

renv: recommended way to control dependencies

```
renv::init()
# Create private package library for project

renv::snapshot()
# Capture dependencies in lockfile

renv::restore()
# Regenerate dependencies from lockfile
```

https://rstudio.github.io/renv/

To ensure reproducibility,

besides R packages

we also need to control

computational environment

`Docker` recreates virtual systems

from a `Dockerfile`

`rang` recreates environment (pkgs + external software)

https://gesistsa.github.io/rang/

GA1: Get the dependency graph of several R packages on CRAN or Github at a specific snapshot date(time)

```
graph ← resolve(c("crsh/papaja", "rio"), snapshot_date = "2019-07-21")
```

Dockerize the dependency graph to a directory

```
dockerize(graph, output_dir = "rangtest")
```

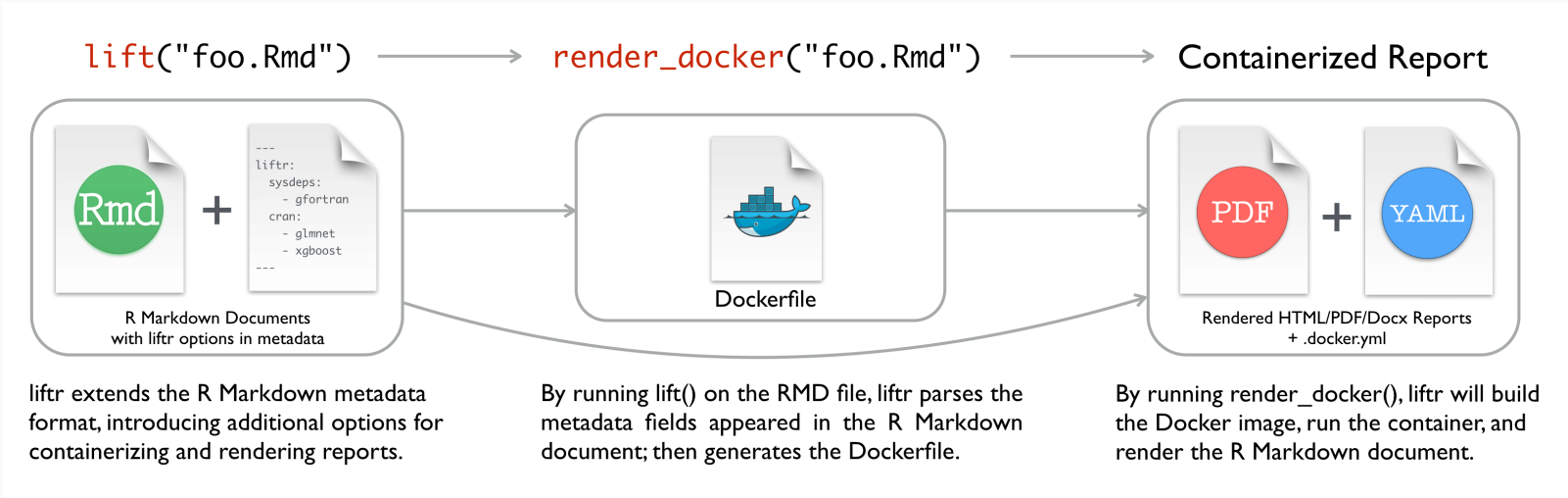
You can build the Docker image either by the R package `stevedore` or Docker CLI client. We use the CLI client.

```
docker build -t rangimg ./rangtest ## might need sudo
```

Launch the container with the built image

```
docker run --rm --name "rangcontainer" -ti rangimg
```

`liftr`: process Rmd in Docker container



https://liftr.me/

`containerit` creates Dockerfile

```
library('containerit')

dockfile <- dockerfile(from = 'mypaper.Rmd')
```

https://o2r.info/containerit

`tugboat` creates Dockerfile w/ entire software environment

tugboat

A simple R package to generate a Dockerfile and corresponding Docker image from an analysis directory. tugboat uses the renv package to automatically detect all the packages necessary to replicate your analysis and will generate a Dockerfile that contains an exact copy of your entire directory with all the packages installed.

tugboat transforms an unstructured analysis folder into a renv.lock file and constructs a Docker image that includes all your essential R packages based on this lockfile.

tugboat may be of use, for example, when preparing a replication package for research. With tugboat, you can take a directory on your local computer and quickly generate a Dockerfile and Docker image that contains all the code and the necessary software to reproduce your findings.

```
library(tugboat)
create()
build()
```

https://www.dmolitor.com/tugboat/

`rix`: reproducible environments with Nix

https://docs.ropensci.org/rix/

> rix is an R package that leverages Nix, a package manager focused on reproducible builds. With Nix, you can create project-specific environments with a custom version of R, its packages, and all system dependencies (e.g., `GDAL`). Nix ensures full reproducibility, which is crucial for research and development projects.

Remember to cite software used!

https://pakillo.github.io/grateful/

```
library('grateful')
cite_packages()
```

`grateful` citation report

R packages used

Package	Version	Citation
base	4.2.3	R Core Team (2023)
lme4	1.1.32	Bates et al. (2015)
tidyverse	2.0.0	Wickham et al. (2019)
vegan	2.6.4	Oksanen et al. (2022)

You can paste this paragraph directly in your report:

We used R version 4.2.3 (R Core Team 2023) and the following R packages: lme4 v. 1.1.32 (Bates et al. 2015), tidyverse v. 2.0.0 (Wickham et al. 2019), vegan v. 2.6.4 (Oksanen et al. 2022).

Package citations

Bates, Douglas, Martin Mächler, Ben Bolker, and Steve Walker. 2015. "Fitting Linear Mixed-Effects Models Using lme4." *Journal of Statistical Software* 67 (1): 1–48. https://doi.org/10.18637/jss.v067.i01.

Oksanen, Jari, Gavin L. Simpson, F. Guillaume Blanchet, Roeland Kindt, Pierre Legendre, Peter R. Minchin, R. B. O'Hara, et al. 2022. *vegan: Community Ecology Package*. https://github.com/vegandevs/vegan.

R Core Team. 2023. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.

Your turn

- Create script/Rmd using different packages
- Call `checkpoint` on former date
- Record dependencies:
  - `renv::snapshot`
- Recreate packages
  - `restore()`