Hands on Workshop

-

Your First Submission in R

Phuse US Connect

Bethesda, 2024

Disclaimer

- This presentation/Workshop represents the views of the authors and not necessarily Novo Nordisk
- Any regulatory communication shown will not be applicable to all projects. Please communicate with the appropriate health authorities about your submission

US Connect 2024

Who are we

Kristian Trøjelsgaard
Statistical Programming Specialist at Novo Nordisk
ktqn@novonordisk.com

Steffen Falgreen Larsen
Principal Statistician at Novo Nordisk
sffl@novonordisk.com

Aim

Collect insight and learnings when using R for regulatory submission

Get an understanding of the requirements and obstacles involved when using R

Motivate discussions between participants (and across companies)

Structure of workshop

	Accuracy	Traceability	Reproducibility
• **Presentation** • Cover some general considerations	✓	✓	✓
• **Exercise** • Group discussion • Arranging pre-made post-it • Reflection on best practices	✓	✓	✓
• **Summary** • Examples of how it can be done	✓	✓	✓

Let's get started

Welcome to Aristotle

A pharmaceutical company that recently did its first submission to FDA which included R programs

Issue regarding submission

Realized that a lot of packages still need **risk-assessments (Accuracy)** and we have gotten questions from FDA on the **Traceability** and **Reproducibility** of our deliverables in R.

As a programmer in this company, we need you to help with this and write up our response to the FDA.

The three essentials

Accuracy
Traceability
Reproducibility

The three essentials

Accuracy

How do we ensure programs are accurate?

- There is a wealth of R packages available
- How do we ensure that the packages that we use are trustworthy?

Traceability

How do we document what we have done?

- Many things in R can influence the final output
  - Packages, Options, Environmental Variables etc.
- What should a log include to provide traceability?

Reproducibility

How to ensure reproducibility when using R

- Package synchronization within a project
- Support authorities in environment replication

Accuracy

How do we ensure accuracy?

R vs SAS: Some fundamental differences

R vs SAS: Some fundamental differences

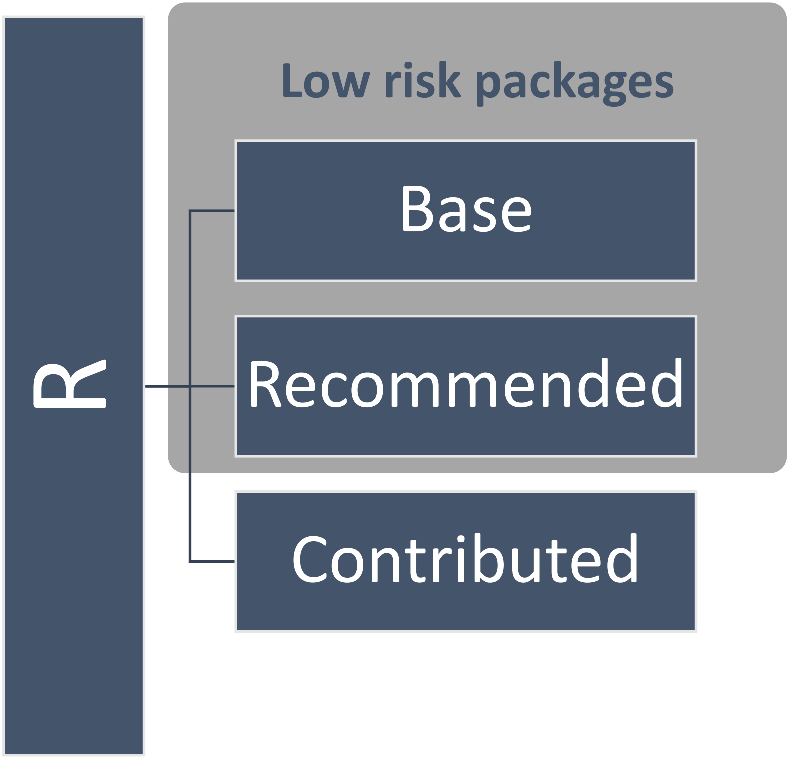
2023:	R 4.3.2	(October, 2023)
	R 4.3.1	(June, 2023)
	R 4.3.0	(April, 2023)
	R 4.2.3	(March, 2023)
2022:	R 4.2.2	(October, 2022)
	R 4.2.1	(June, 2022)
	R 4.2.0	(April, 2022)
	R 4.1.3	(March, 2022)
2021:	R 4.1.2	(November, 2021)
	R 4.1.1	(August, 2021)
	R 4.1.0	(May, 2021)
	R 4.0.5	(March, 2021)
	R 4.0.4	(February, 2021)
2020:	R 4.0.3	(October, 2020)
	R 4.0.2	(June, 2020)
	R 4.0.1	(June, 2020)
	R 4.0.0	(April, 2020)
	R 3.6.3	(February, 2020)

In Novo Nordisk

2023:	SAS 9.4M8	(Jan, 2023)
2020:	SAS 9.4M7	(August, 2020)
2018:	SAS 9.4M6	(November, 2018)
2017:	SAS 9.4M5	(September, 2017
2016:	SAS 9.4M4	(November, 2016)
2015:	SAS 9.4M3	(July, 2015)
2014:	SAS 9.4M2	(August, 2014)
2013:	SAS 9.4M1	(December, 2013)
	SAS 9.4	(July, 2013)

R vs SAS: Some fundamental differences

CRAN



- **Base/Recommended**: (Can be trusted)
  - The R Foundation develops both the base and recommended packages, and follows practices that ensures the accuracy of each

- **Contributed**: (Need internal testing)
  - 20.000+ **R** packages on CRAN, all tested to some extent, but not all can be considered validated

More info in this white-paper from the R validation hub

Contributed

CRAN checks include (but is not limited to):

Execution of all test

Running all examples

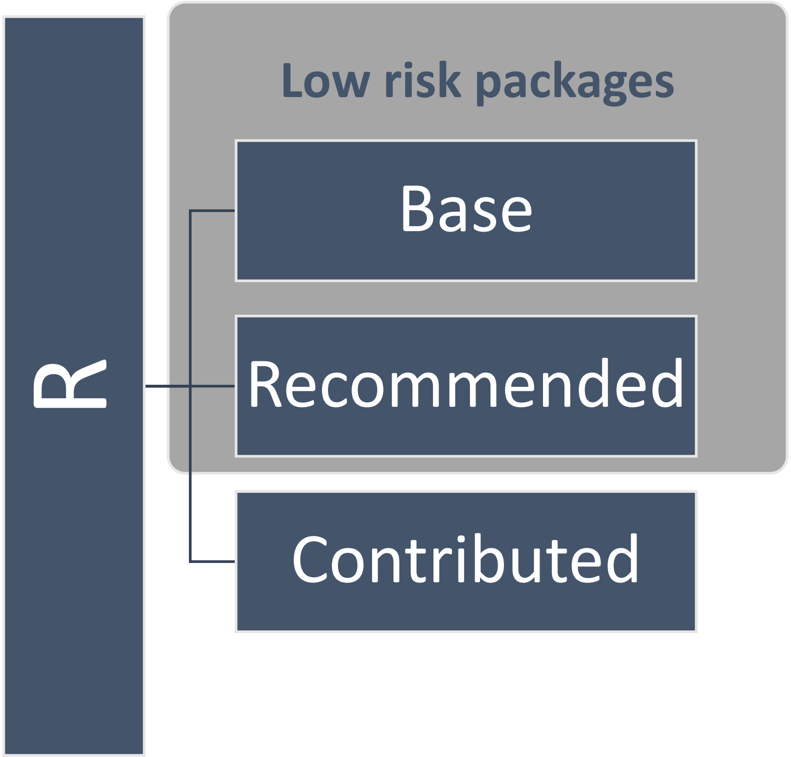
Checking dependencies

Verifying documentation

Looking for syntax errors

Re-building the vignettes

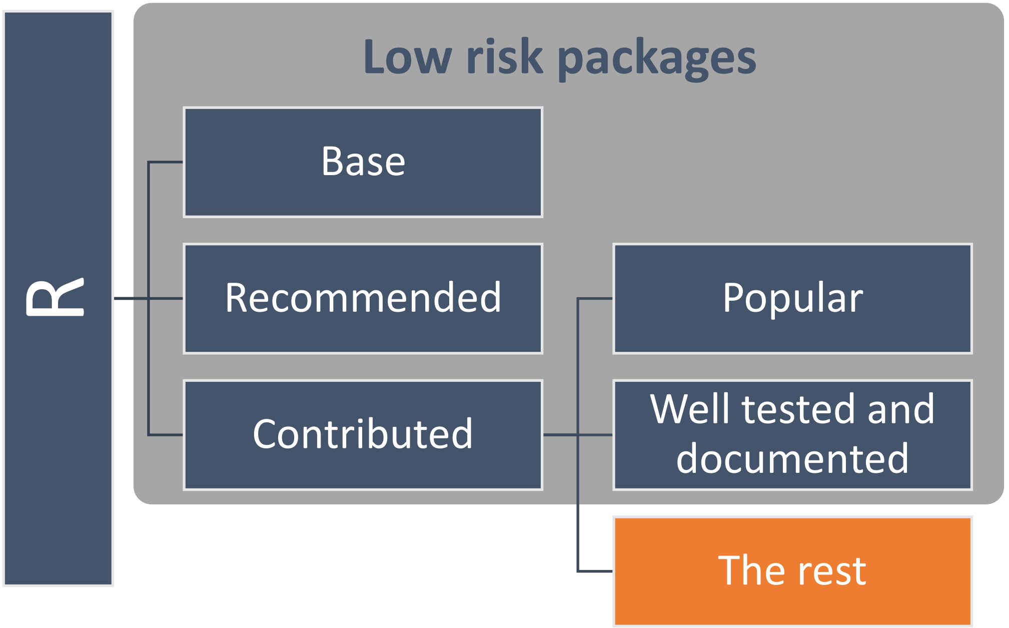
CRAN



- **Base/Recommended**: (Can be trusted)
  - The R Foundation develops both the base and recommended packages, and follows practices that ensures the accuracy of each
- **Contributed**: (Need internal testing)
  - 20.000+ **R** packages on CRAN, all tested to some extent, but not all can be considered validated

More info in this white-paper from the R validation hub

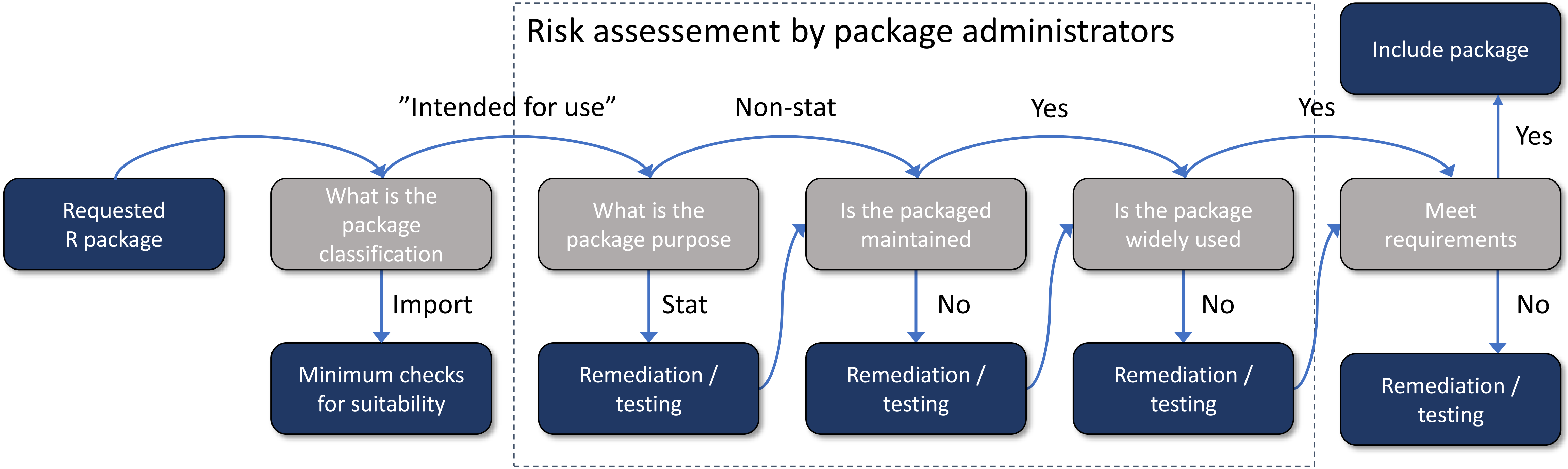
CRAN



- **Base/Recommended**: (Can be trusted)
  - The R Foundation develops both the base and recommended packages, and follows practices that ensures the accuracy of each
- **Contributed**: (Need internal testing)
  - 20.000+ **R** packages on CRAN, all tested to some extent, but not all can be considered validated
- **Popular**: (Very low risk)
  - A subset of the contributed packages have an extremely large userbase and/or extensive test-suites
  - tidyverse, data.table, …

More info in this white-paper from the R validation hub

Risk assessment



Risk assessement by package administrators

Requested R package

"Intended for use"

What is the package classification

Import

Minimum checks for suitability

Non-stat

What is the package purpose

Stat

Remediation / testing

Yes

Is the packaged maintained

No

Remediation / testing

Yes

Is the package widely used

No

Remediation / testing

Meet requirements

Yes

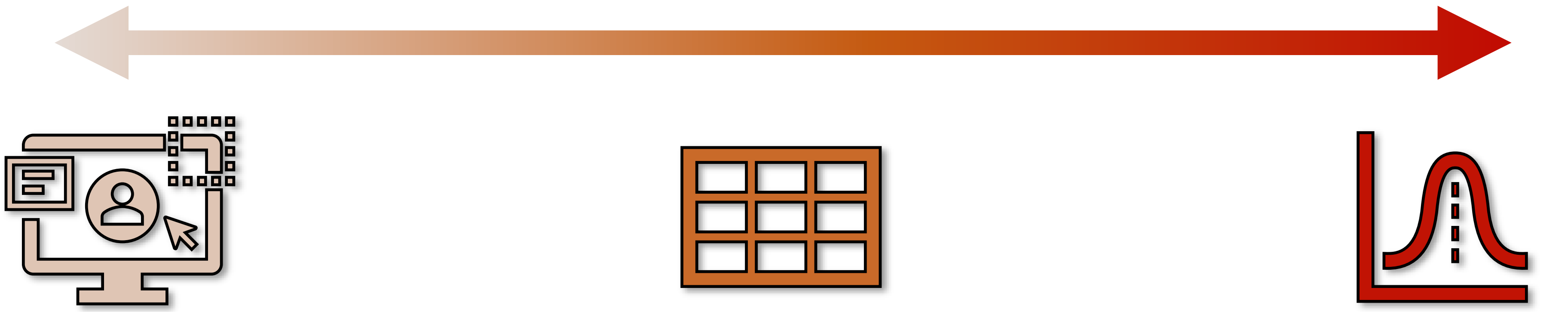
Include package

No

Remediation / testing

What is the package purpose

The **risk** of including a package is related to the **usage**



Low risk

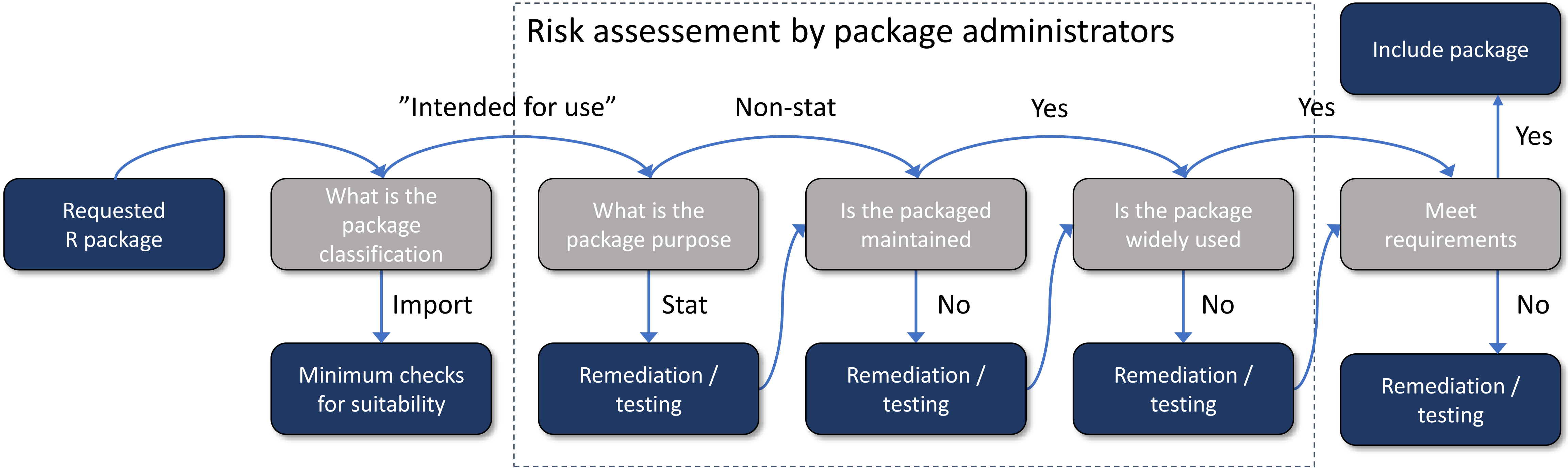
High risk

Homepage generation

Data manipulation

Statistical analysis

Risk assessment



Risk assessement by package administrators

Is the packaged maintained

- Number of contributors
- Reputation

Are issues solved at a reasonable pace

Pace of new releases

Last update

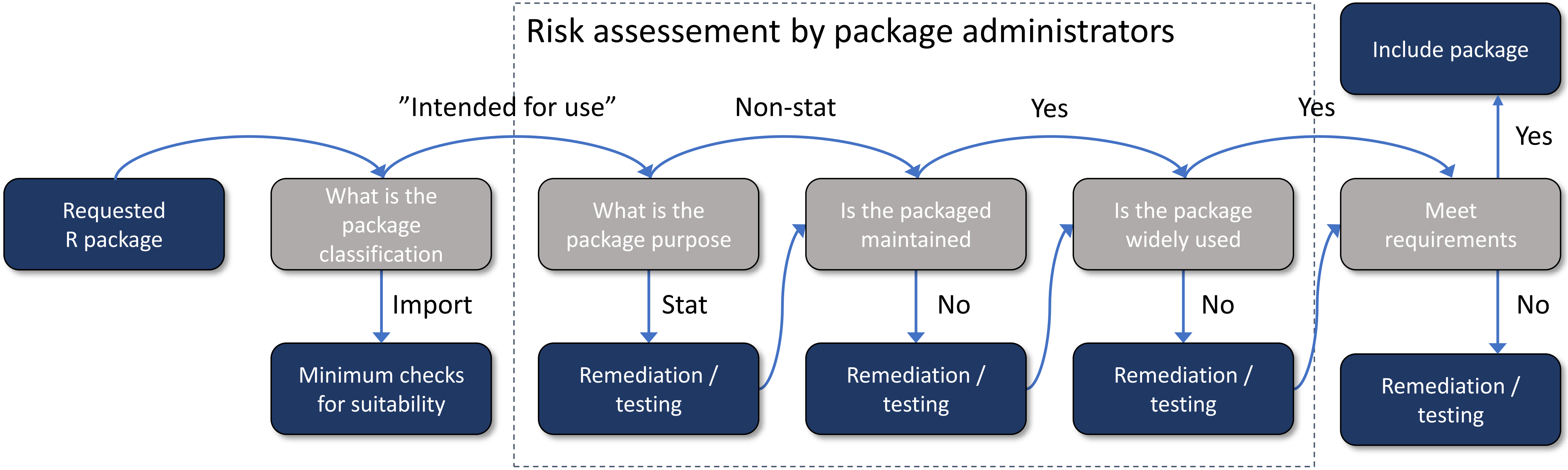
Testing
(code coverage)

Vignettes

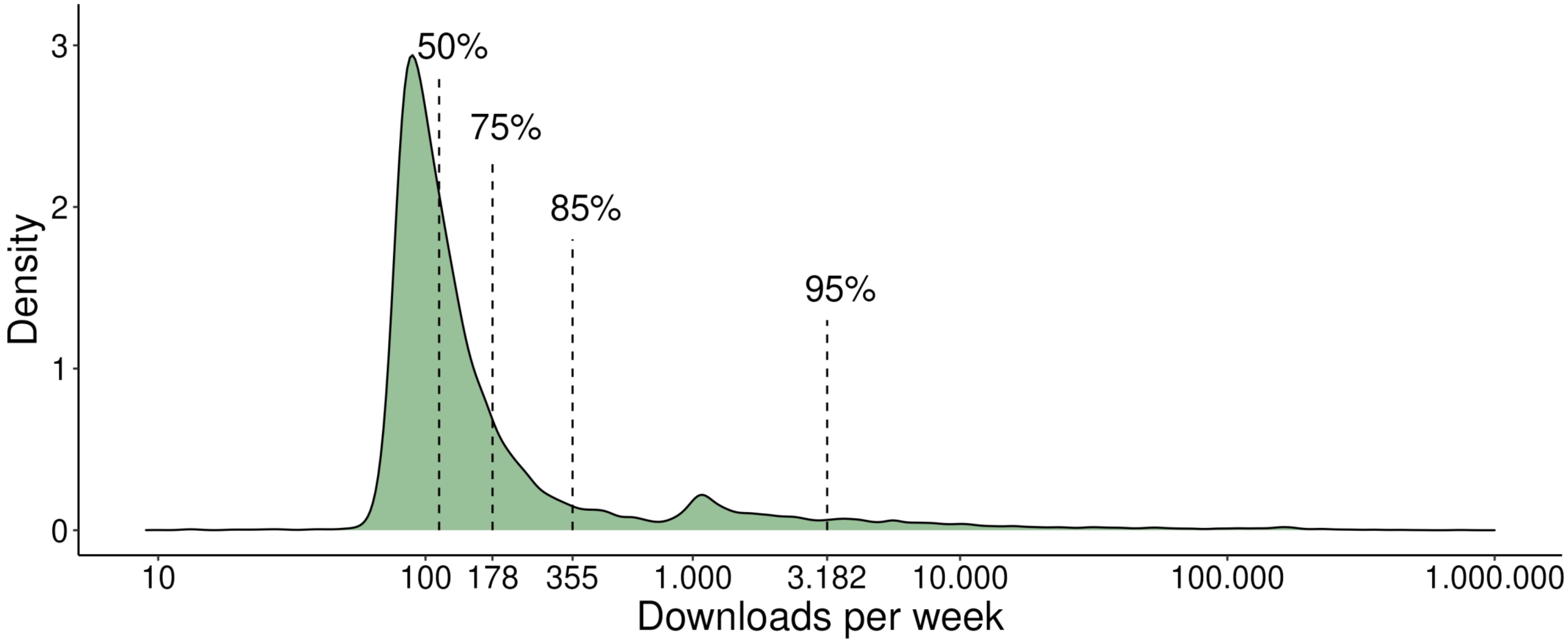
Does all functions have examples

News for each release

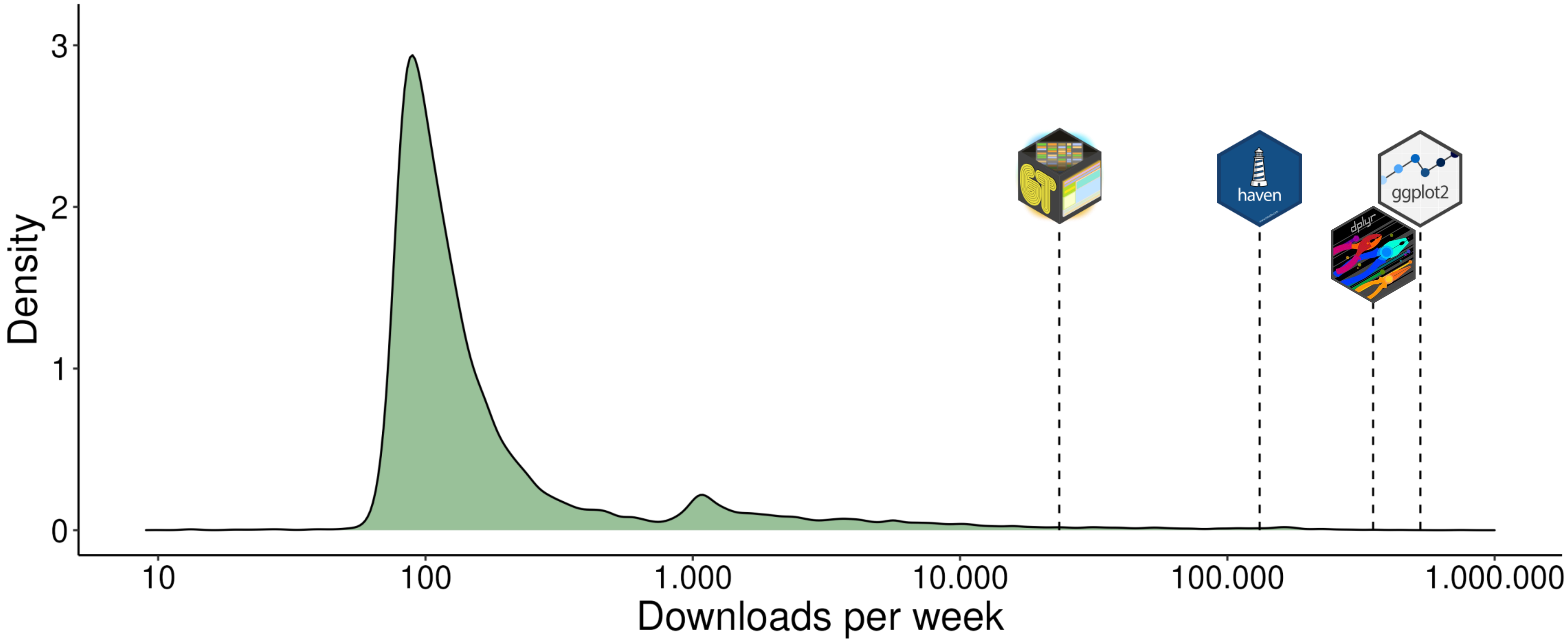
Risk assessment



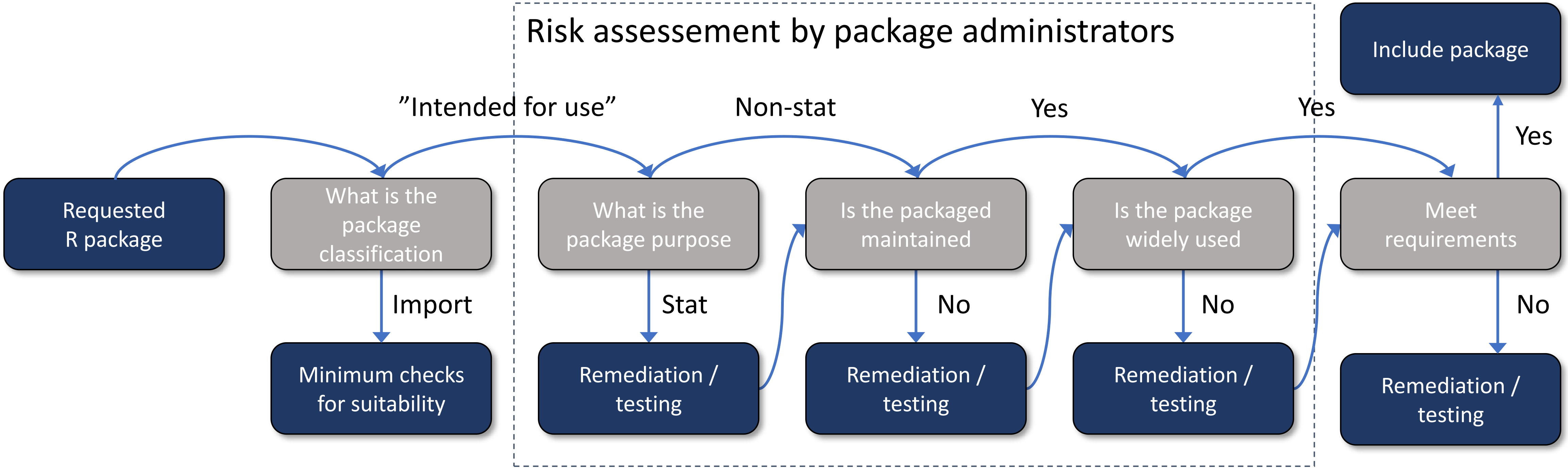
Is the package widely used



Is the package widely used



Risk assessment



Risk assesement by package administrators

Obtaining metrics

Metrics

dplyr

First on CRAN:	2014-01-29
Last update:	2023-11-17
Contributors:	> 250
Code coverage:	84 %
New releases:	3-4 times a year
Downloads:	Approx. 350.000 per week
Examples:	95% of the functions have examples

stringr

First on CRAN:	2009-11-09
Last update:	2023-11-14
Contributors:	> 50
Code coverage:	99 %
New releases:	1-2 times a year
Downloads:	Approx. 260.000 per week
Examples:	100% of the functions have examples

Accuracy

Exercise

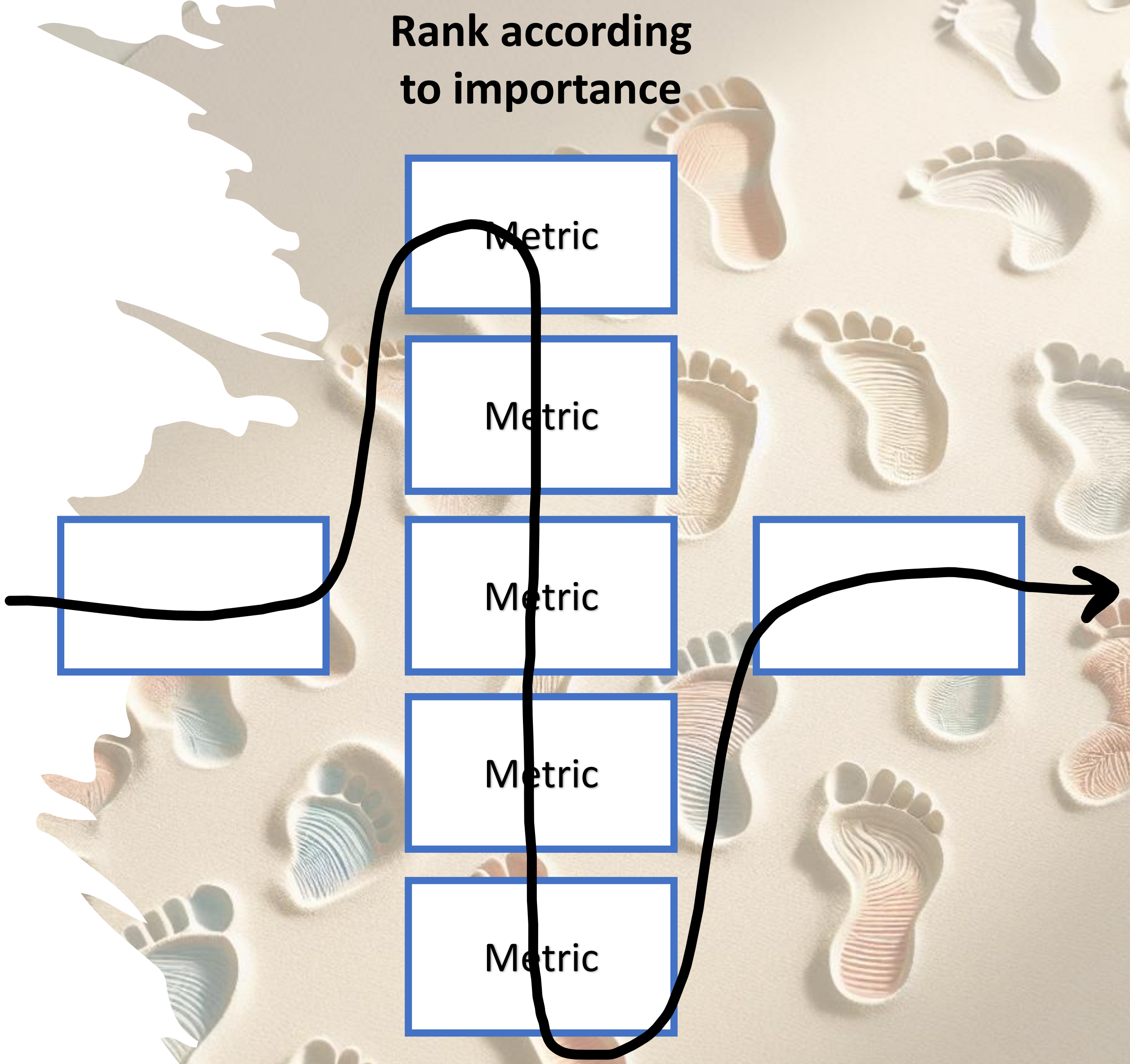
Exercise

Build workflow:

- Build your own assessment work-flow based on the handouts
- Add additional steps if you believe anything is missing

Assess packages:

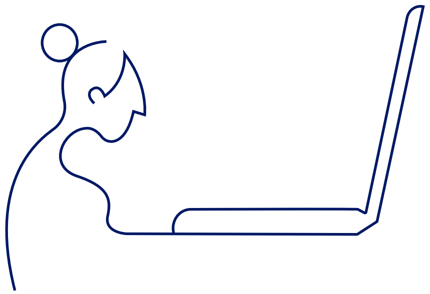
- Assess the example packages

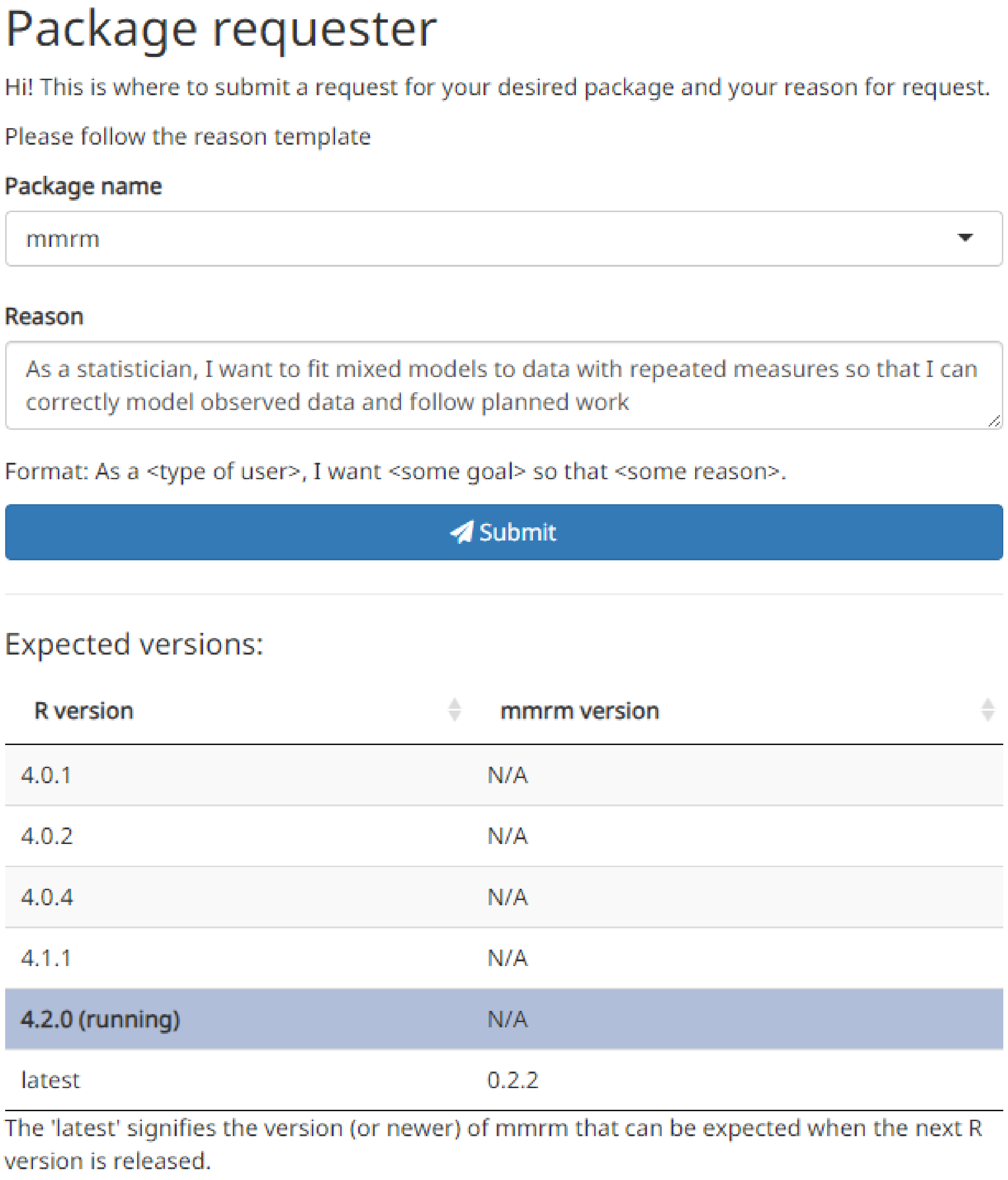


Accuracy

Summary
&
Example from Novo Nordisk

Package requests





Package requester

Hi! This is where to submit a request for your desired package and your reason for request.

Please follow the reason template

Package name

mmrm

Reason

As a statistician, I want to fit mixed models to data with repeated measures so that I can correctly model observed data and follow planned work

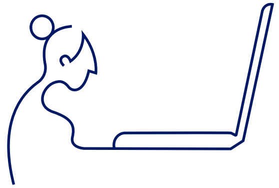
Format: As a <type of user>, I want <some goal> so that <some reason>.

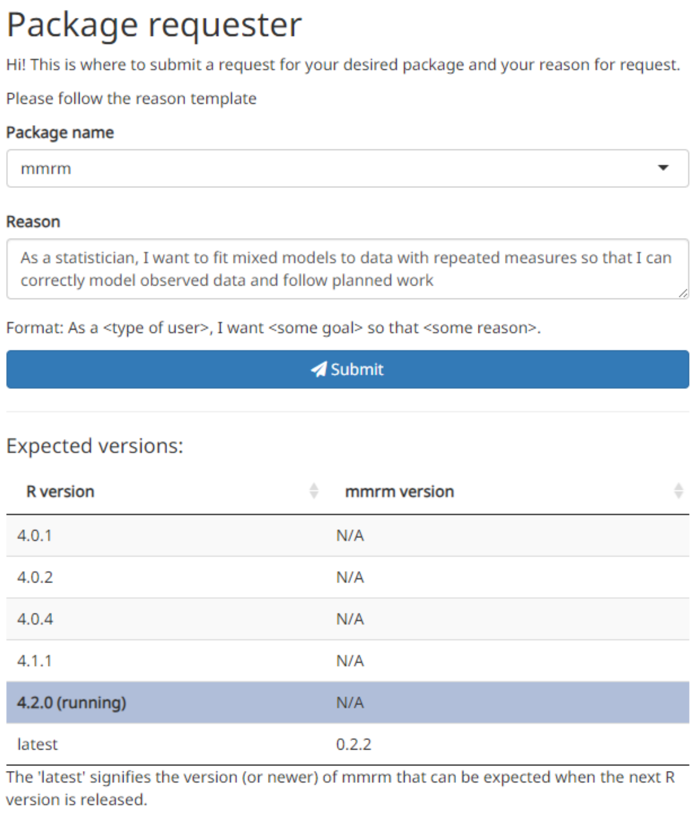
Expected versions:

R version	mmrm version
4.0.1	N/A
4.0.2	N/A
4.0.4	N/A
4.1.1	N/A
4.2.0 (running)	N/A
latest	0.2.2

The 'latest' signifies the version (or newer) of mmrm that can be expected when the next R version is released.

Package requests





Package requester

Hi! This is where to submit a request for your desired package and your reason for request.

Please follow the reason template

Package name

mmrm

Reason

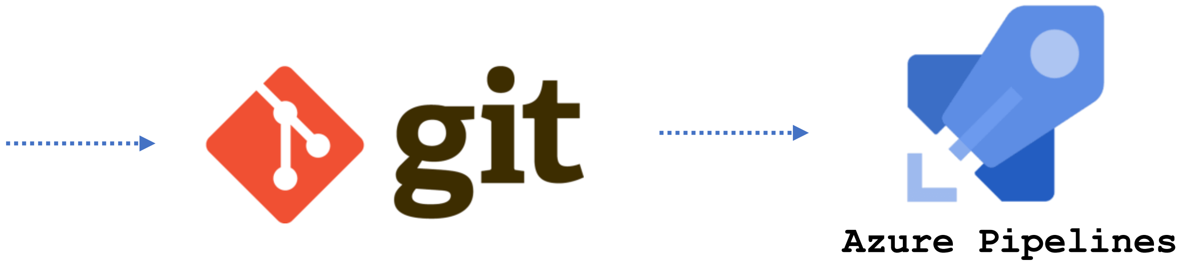
As a statistician, I want to fit mixed models to data with repeated measures so that I can correctly model observed data and follow planned work

Format: As a <type of user>, I want <some goal> so that <some reason>.

Expected versions:

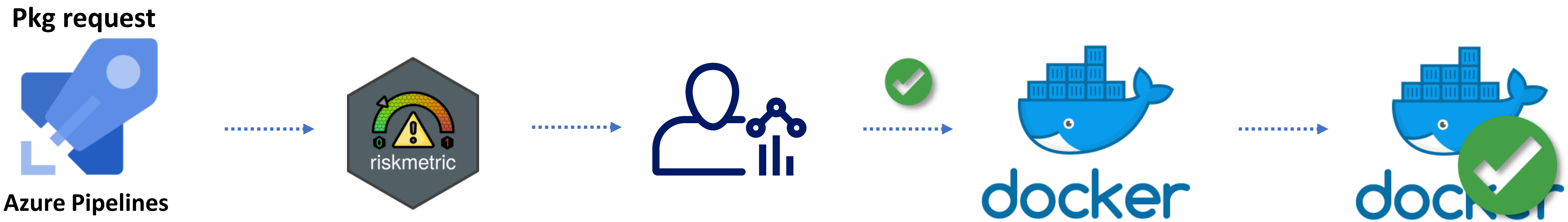
R version	mmrm version
4.0.1	N/A
4.0.2	N/A
4.0.4	N/A
4.1.1	N/A
4.2.0 (running)	N/A
latest	0.2.2

The 'latest' signifies the version (or newer) of mmrm that can be expected when the next R version is released.



Package risk assessment

For every package requested by a user continuous integration is used to evaluate the package



Pkg request

Azure Pipelines

Risk assessment:

- A reviewer **manually** inspects results
- **Approve** or **Reject**
- Document decision

Release:

- If approved the package is installed in the next image

Check:

- Running all **R-CMD** checks on the docker image and compare against human approved versions
- **Additional tests** also executed

The three essentials

Accuracy

How do we ensure programs are accurate?

Traceability

How do we document what we have done?

Reproducibility

How to ensure reproducibility when using R

The three essentials

Accuracy

How do we ensure programs are accurate?

Traceability

How do we document what we have done?

- Many things in R can influence the final output
  - Packages, Options, Environmental Variables etc.
- What should a log include to provide traceability?

Reproducibility

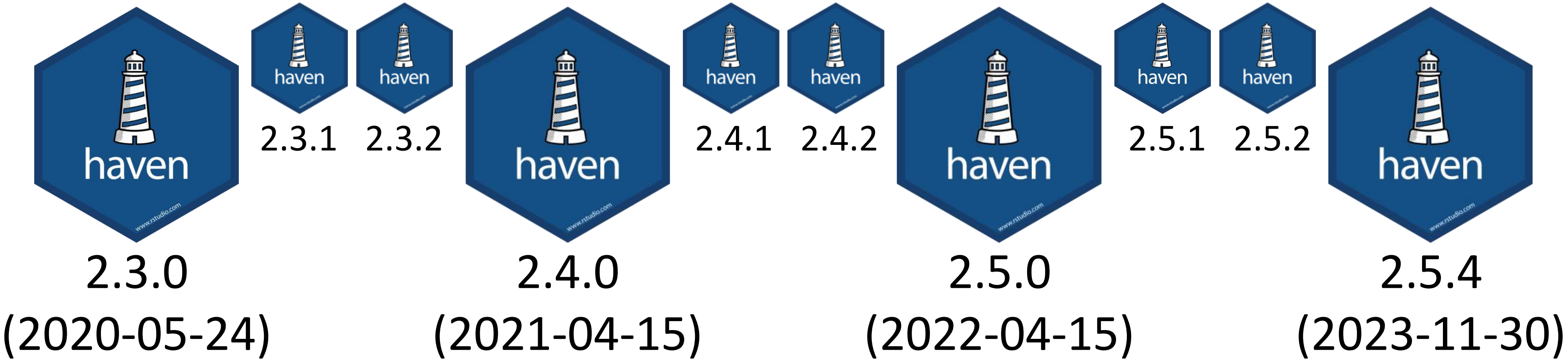
How to ensure reproducibility when using R

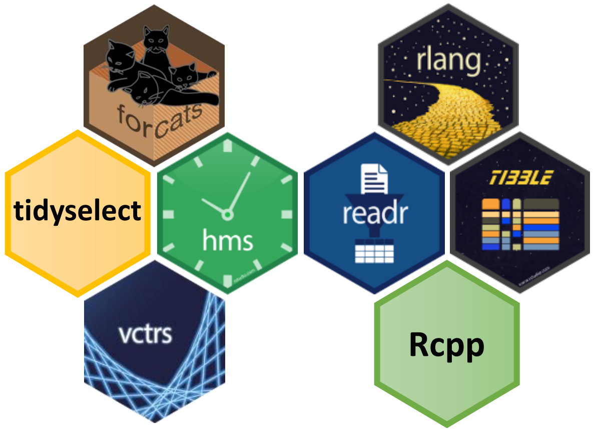
R and all the packages

test that
Shiny
devtools
haven
covr
dplyr
ggplot2
usethis
THIS SIDE UP
pkgdown
R 2
roxygen2
renv

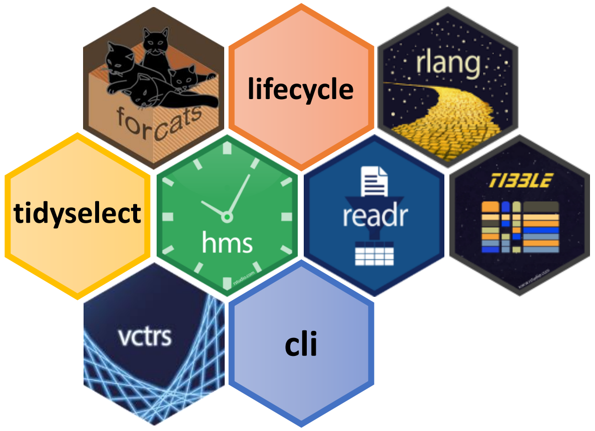
The essentials: Traceability

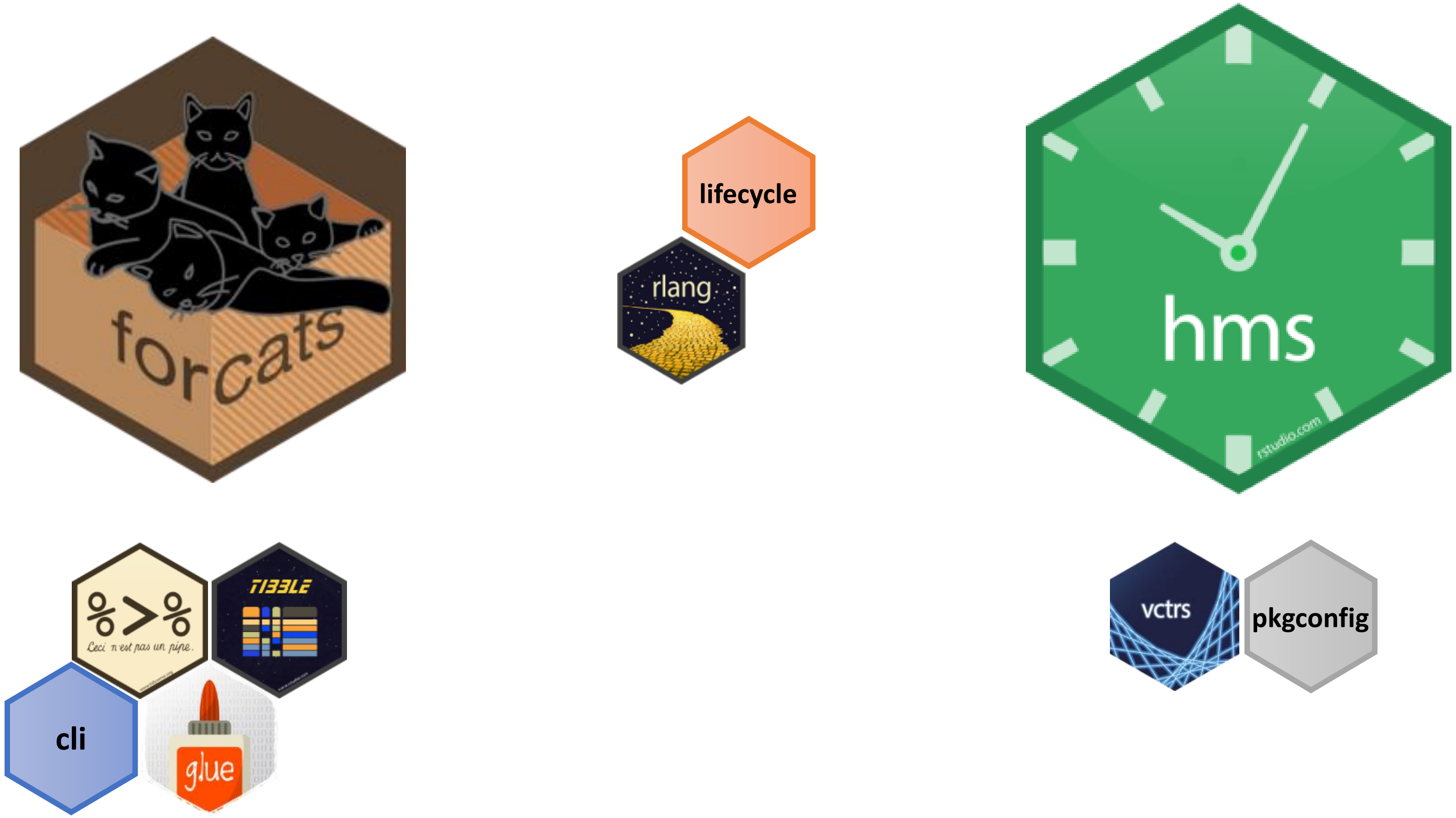
Each package usually exist in many different versions:





Multiple dependencies




forcats
lifecycle
rlang
hms
rstudio.com
%>%
Ceci n'est pas un pipe.
TIBBLE
cli
glue
vctrs
pkgconfig

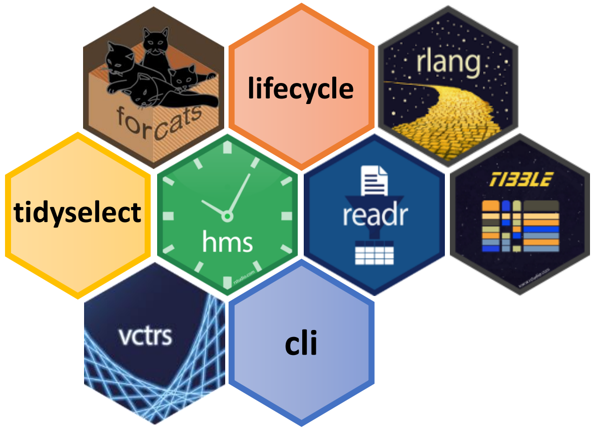
Attached vs Imports.

Attached

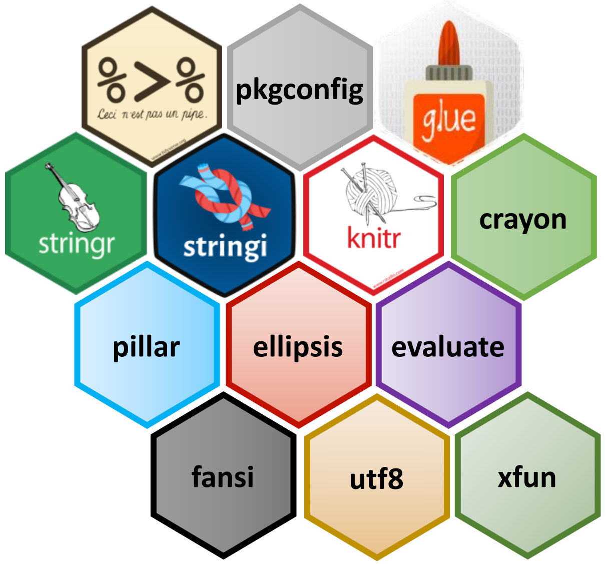
haven
www.rstudio.com

```
library("haven")
```

Imports



Imports



Options and Environment variables

- Options and environment variables can influence the behavior of the packages.

Many packages rely heavily on R options. Examples could be:

- Rules for rounding
- Contrasts: What is the default reference when comparing factors in statistical analysis
- Burn-in in Bayesian statistical modelling
- Some statistical packages use options to determine accuracy and/or number of runs

Environment variables can also modify the outcome of R code:

- Some R options might be modified through environment variables.
- Package can also rely directly on environment variables
- E.g. {renv} rely on environment variables when looking for R packages

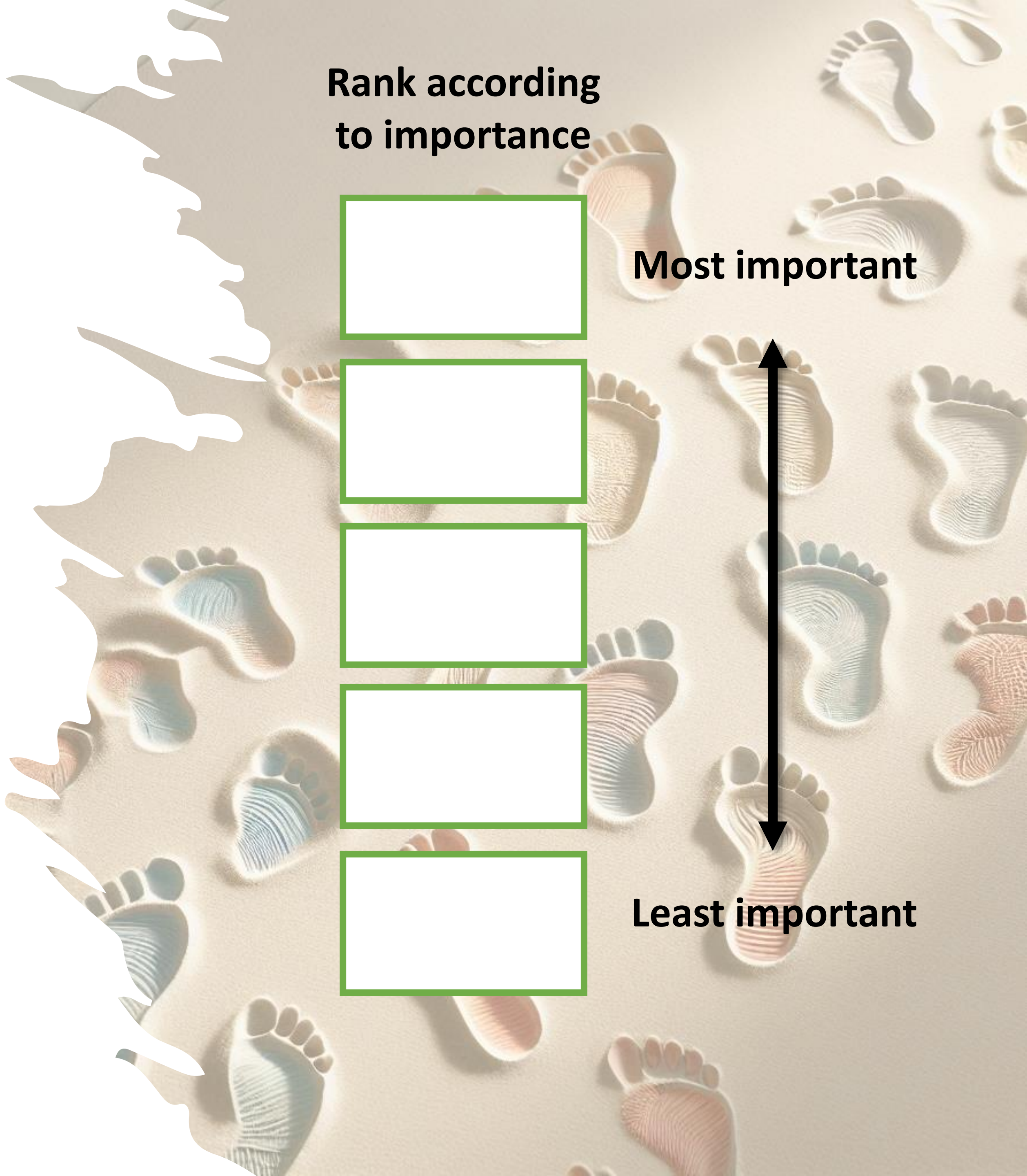
Traceability

Exercise

Exercise

Purpose:

- Create a log for R scripts
- Prioritize content according to criticality
- Create your own content if something vital and/or nice is missing



Rank according to importance

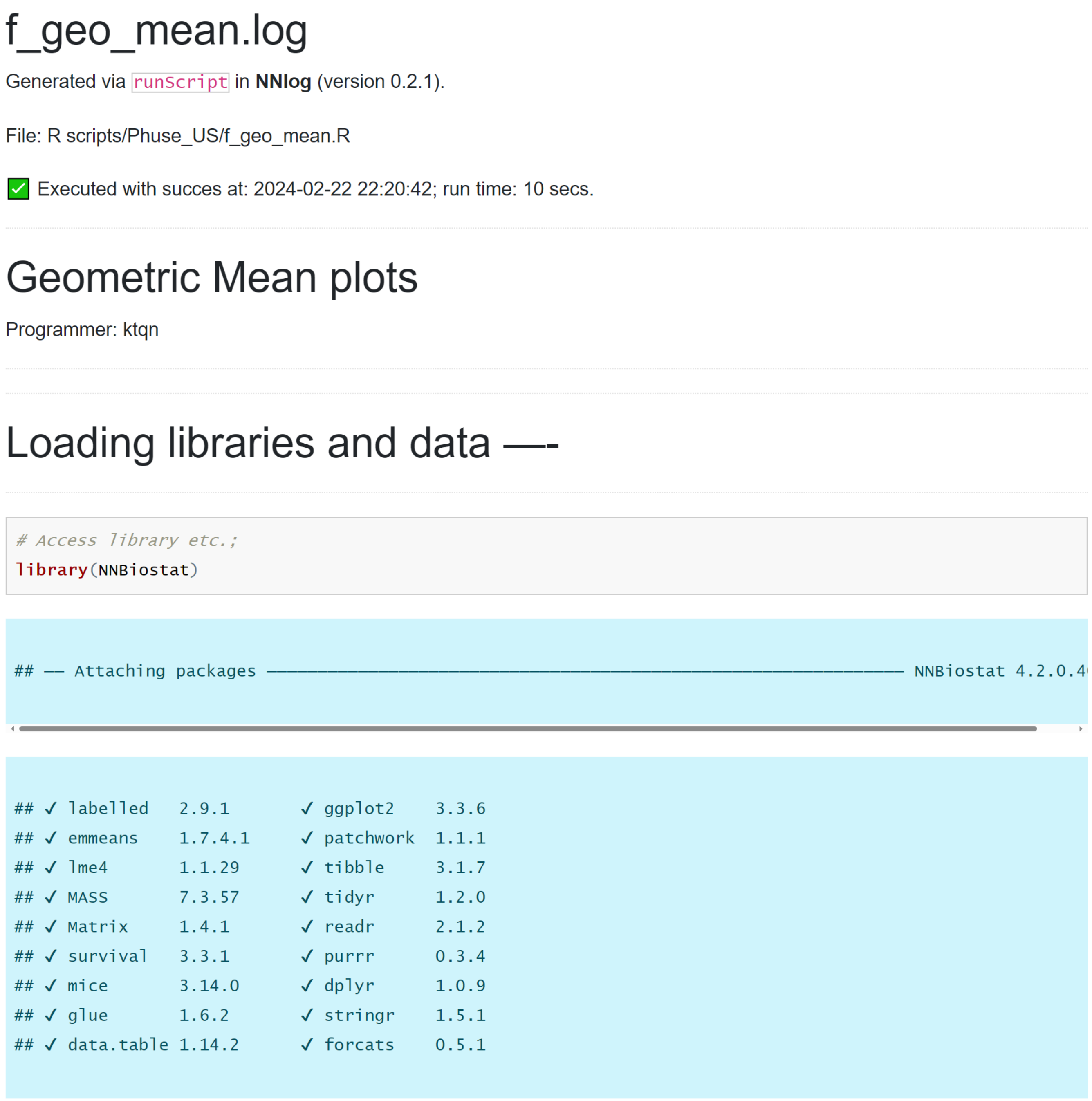
Most important

Least important

Traceability

Summary
&
Example from Novo Nordisk

General info



f_geo_mean.log

Generated via `runScript` in **NNlog** (version 0.2.1).

File: R scripts/Phuse_US/f_geo_mean.R

✅ Executed with succes at: 2024-02-22 22:20:42; run time: 10 secs.

Geometric Mean plots

Programmer: ktqn

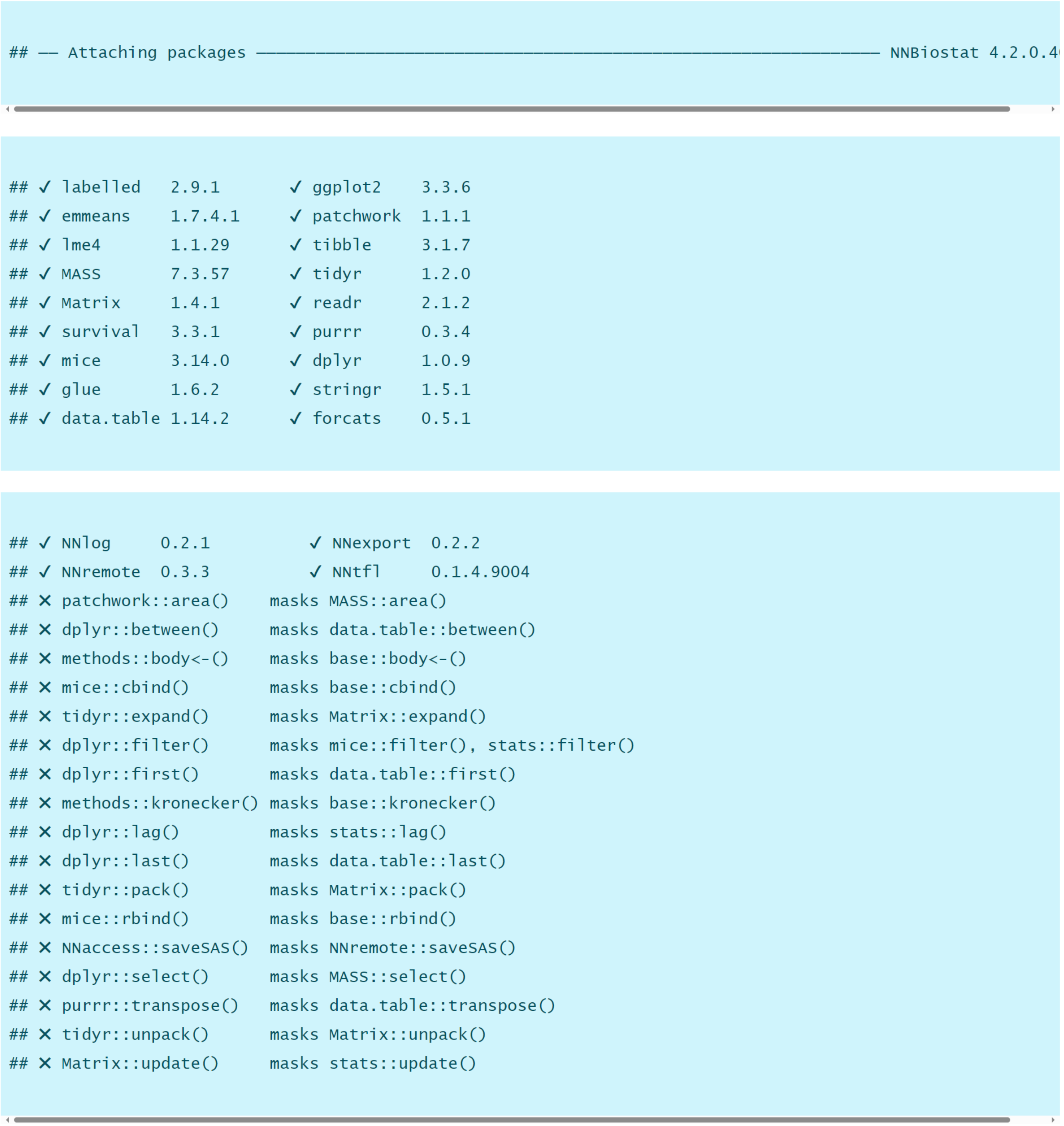
Loading libraries and data —-

```
# Access library etc.;
library(NNBiostat)
```

```
## —— Attaching packages ——————————————————————————————————————————————————————————— NNBiostat 4.2.0.4
```

```
## ✓ labelled   2.9.1      ✓ ggplot2   3.3.6
## ✓ emmeans    1.7.4.1    ✓ patchwork 1.1.1
## ✓ lme4       1.1.29     ✓ tibble    3.1.7
## ✓ MASS       7.3.57     ✓ tidyr     1.2.0
## ✓ Matrix     1.4.1      ✓ readr     2.1.2
## ✓ survival   3.3.1      ✓ purrr     0.3.4
## ✓ mice       3.14.0     ✓ dplyr     1.0.9
## ✓ glue       1.6.2      ✓ stringr   1.5.1
## ✓ data.table 1.14.2     ✓ forcats   0.5.1
```

Masked functions

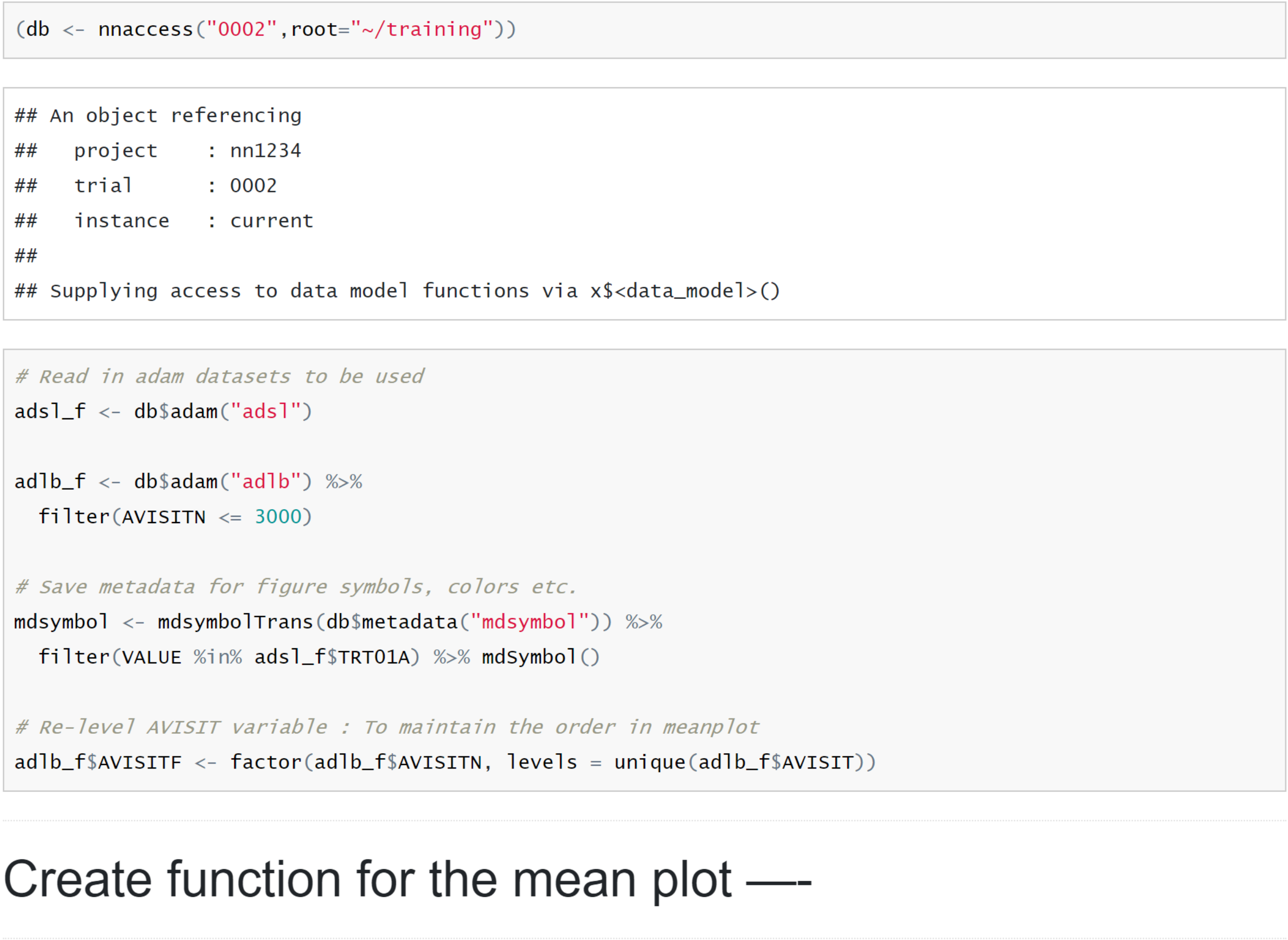


```
## ── Attaching packages ─────────────────────────────────────────────────────── NNBiostat 4.2.0.4
```

```
## ✓ labelled   2.9.1       ✓ ggplot2   3.3.6
## ✓ emmeans    1.7.4.1     ✓ patchwork 1.1.1
## ✓ lme4       1.1.29      ✓ tibble    3.1.7
## ✓ MASS       7.3.57      ✓ tidyr     1.2.0
## ✓ Matrix     1.4.1       ✓ readr     2.1.2
## ✓ survival   3.3.1       ✓ purrr     0.3.4
## ✓ mice       3.14.0      ✓ dplyr     1.0.9
## ✓ glue       1.6.2       ✓ stringr   1.5.1
## ✓ data.table 1.14.2      ✓ forcats   0.5.1
```

```
## ✓ NNlog     0.2.1          ✓ NNexport  0.2.2
## ✓ NNremote  0.3.3          ✓ NNtfl     0.1.4.9004
## ✕ patchwork::area()    masks MASS::area()
## ✕ dplyr::between()     masks data.table::between()
## ✕ methods::body<-()    masks base::body<-()
## ✕ mice::cbind()        masks base::cbind()
## ✕ tidyr::expand()      masks Matrix::expand()
## ✕ dplyr::filter()      masks mice::filter(), stats::filter()
## ✕ dplyr::first()       masks data.table::first()
## ✕ methods::kronecker() masks base::kronecker()
## ✕ dplyr::lag()         masks stats::lag()
## ✕ dplyr::last()        masks data.table::last()
## ✕ tidyr::pack()        masks Matrix::pack()
## ✕ mice::rbind()        masks base::rbind()
## ✕ NNaccess::saveSAS()  masks NNremote::saveSAS()
## ✕ dplyr::select()      masks MASS::select()
## ✕ purrr::transpose()   masks data.table::transpose()
## ✕ tidyr::unpack()      masks Matrix::unpack()
## ✕ Matrix::update()     masks stats::update()
```

Code chunks



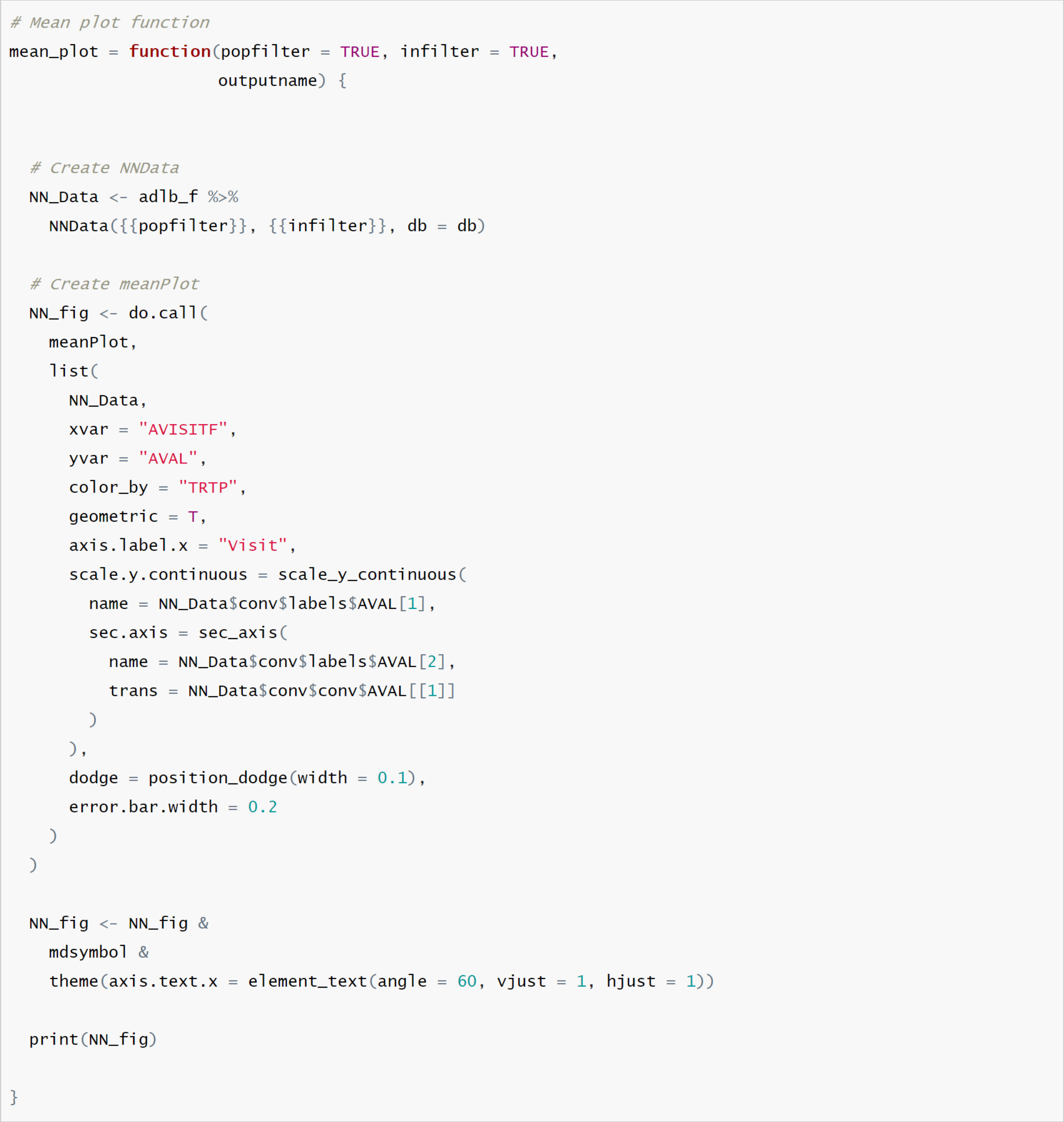
```
(db <- nnaccess("0002",root="~/training"))
```

```
## An object referencing
##   project    : nn1234
##   trial      : 0002
##   instance   : current
##
## Supplying access to data model functions via x$<data_model>()
```

```
# Read in adam datasets to be used
adsl_f <- db$adam("adsl")

adlb_f <- db$adam("adlb") %>%
  filter(AVISITN <= 3000)

# Save metadata for figure symbols, colors etc.
mdsymbol <- mdsymbolTrans(db$metadata("mdsymbol")) %>%
  filter(VALUE %in% adsl_f$TRT01A) %>% mdSymbol()

# Re-level AVISIT variable : To maintain the order in meanplot
adlb_f$AVISITF <- factor(adlb_f$AVISITN, levels = unique(adlb_f$AVISIT))
```

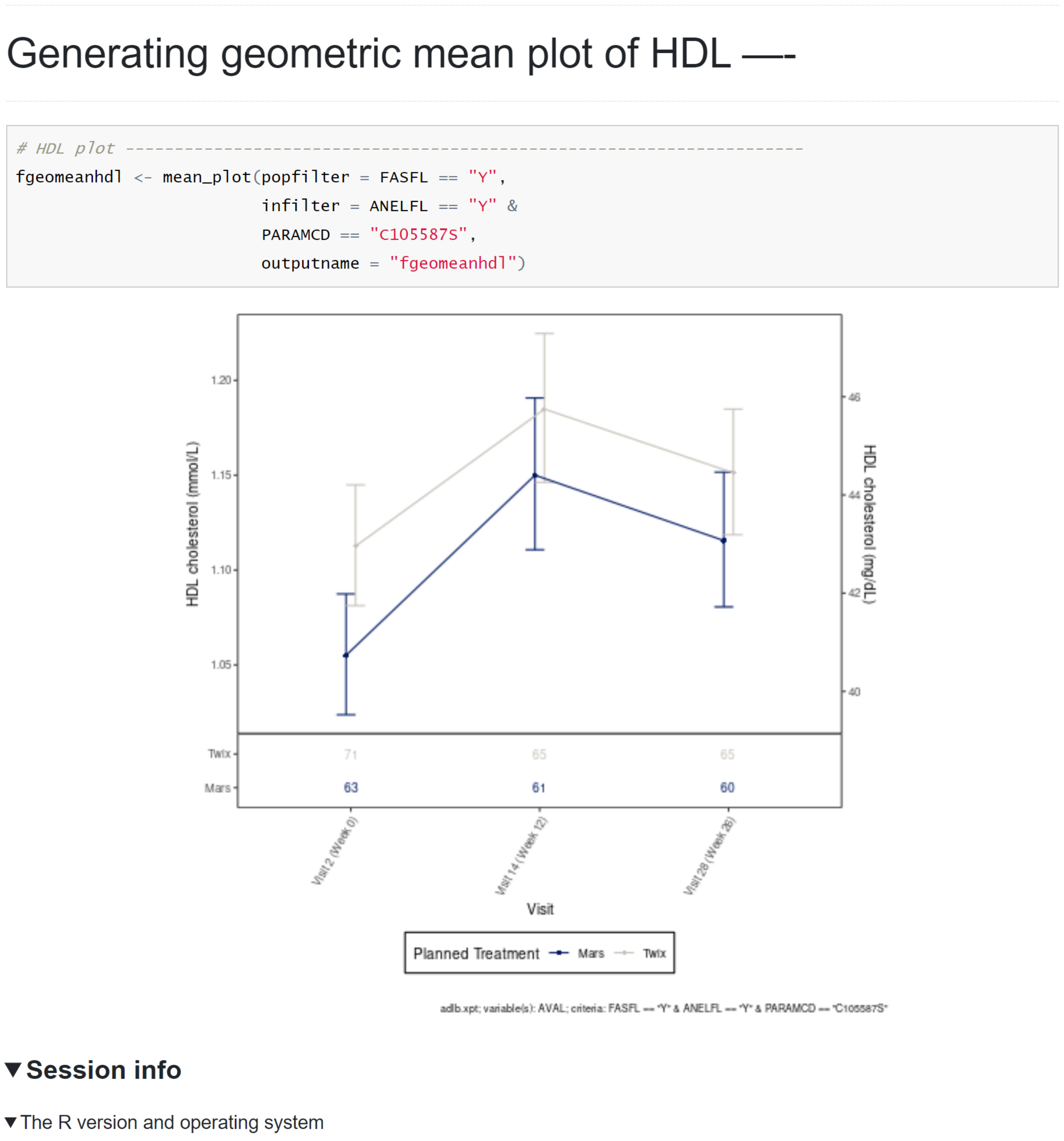
Create function for the mean plot —-

Creating a function



```r
# Mean plot function
mean_plot = function(popfilter = TRUE, infilter = TRUE,
                     outputname) {



  # Create NNData
  NN_Data <- adlb_f %>%
    NNData({{popfilter}}, {{infilter}}, db = db)

  # Create meanPlot
  NN_fig <- do.call(
    meanPlot,
    list(
      NN_Data,
      xvar = "AVISITF",
      yvar = "AVAL",
      color_by = "TRTP",
      geometric = T,
      axis.label.x = "Visit",
      scale.y.continuous = scale_y_continuous(
        name = NN_Data$conv$labels$AVAL[1],
        sec.axis = sec_axis(
          name = NN_Data$conv$labels$AVAL[2],
          trans = NN_Data$conv$conv$AVAL[[1]]
        )
      ),
      dodge = position_dodge(width = 0.1),
      error.bar.width = 0.2
    )
  )

  NN_fig <- NN_fig &
    mdsymbol &
    theme(axis.text.x = element_text(angle = 60, vjust = 1, hjust = 1))

  print(NN_fig)

}
```

Execute function



Generating geometric mean plot of HDL —-

```
# HDL plot ---------------------------------------------------------------------
fgeomeanhdl <- mean_plot(popfilter = FASFL == "Y",
                         infilter = ANELFL == "Y" &
                         PARAMCD == "C105587S",
                         outputname = "fgeomeanhdl")
```

adlb.xpt; variable(s): AVAL; criteria: FASFL == "Y" & ANELFL == "Y" & PARAMCD == "C105587S"

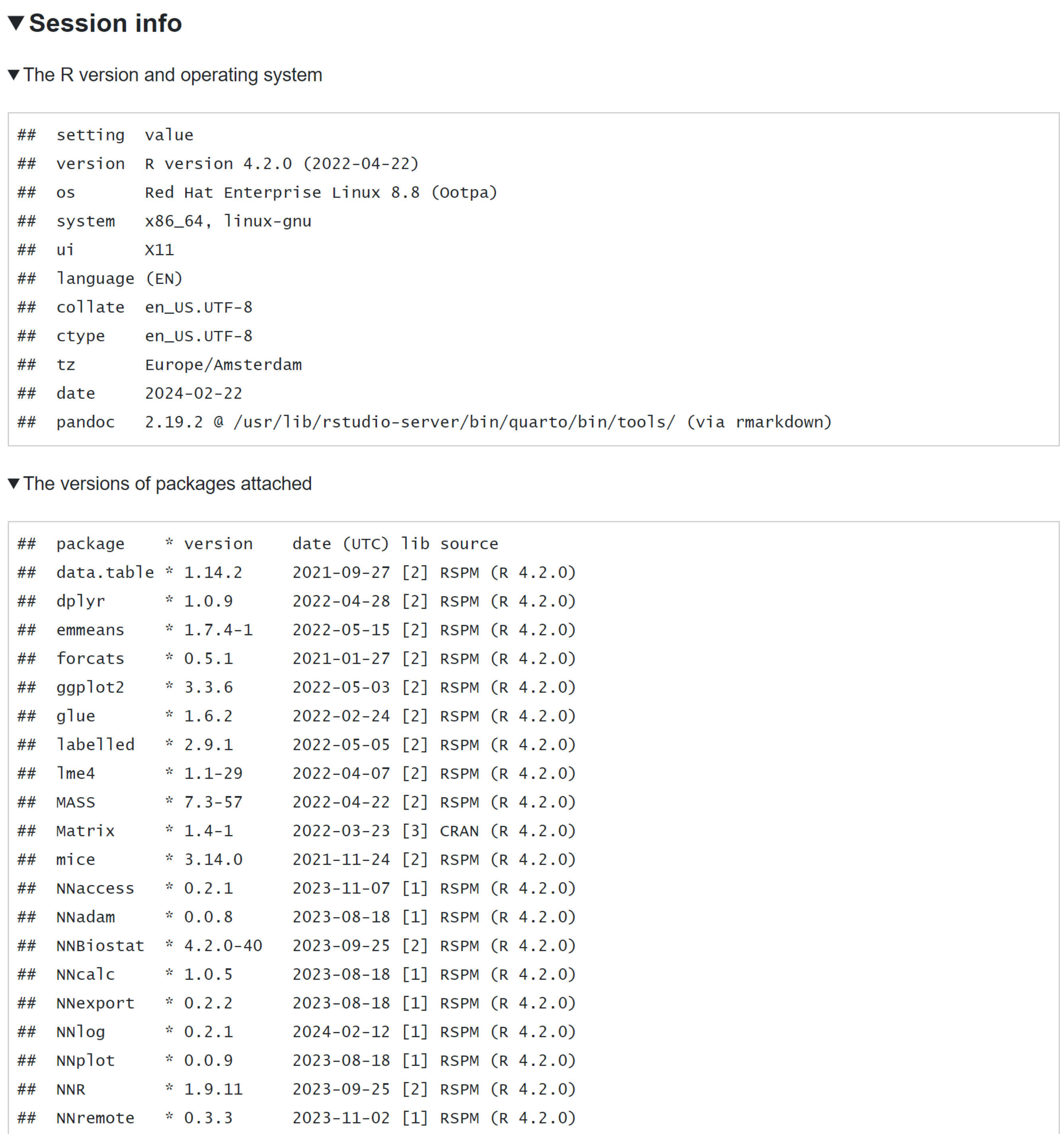
▼Session info

▼The R version and operating system

R version

OS

Packages



▼Session info

▼The R version and operating system

```
##  setting  value
##  version  R version 4.2.0 (2022-04-22)
##  os       Red Hat Enterprise Linux 8.8 (Ootpa)
##  system   x86_64, linux-gnu
##  ui       X11
##  language (EN)
##  collate  en_US.UTF-8
##  ctype    en_US.UTF-8
##  tz       Europe/Amsterdam
##  date     2024-02-22
##  pandoc   2.19.2 @ /usr/lib/rstudio-server/bin/quarto/bin/tools/ (via rmarkdown)
```

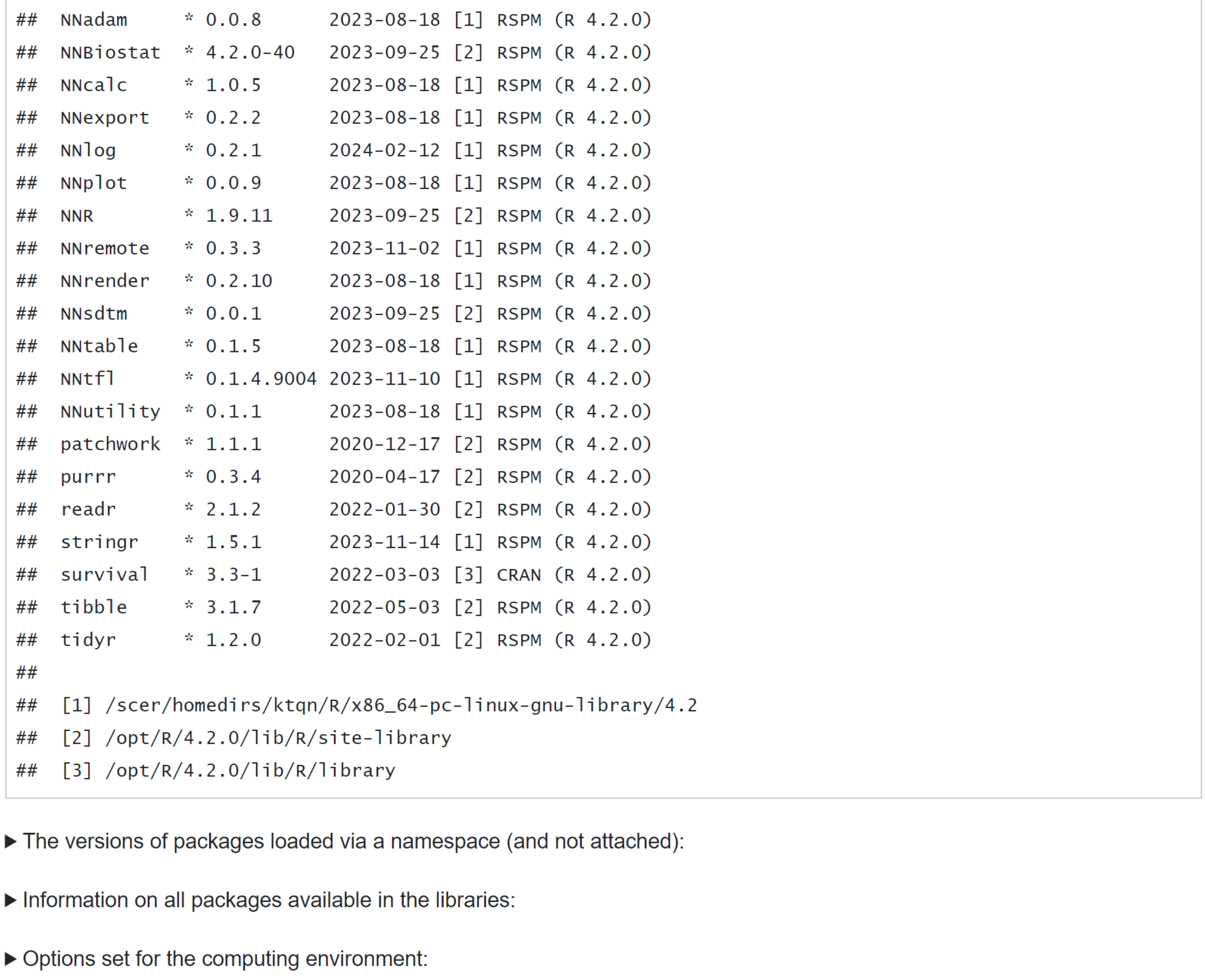
▼The versions of packages attached

```
##  package    * version    date (UTC) lib source
##  data.table * 1.14.2     2021-09-27 [2] RSPM (R 4.2.0)
##  dplyr      * 1.0.9      2022-04-28 [2] RSPM (R 4.2.0)
##  emmeans    * 1.7.4-1    2022-05-15 [2] RSPM (R 4.2.0)
##  forcats    * 0.5.1      2021-01-27 [2] RSPM (R 4.2.0)
##  ggplot2    * 3.3.6      2022-05-03 [2] RSPM (R 4.2.0)
##  glue       * 1.6.2      2022-02-24 [2] RSPM (R 4.2.0)
##  labelled   * 2.9.1      2022-05-05 [2] RSPM (R 4.2.0)
##  lme4       * 1.1-29     2022-04-07 [2] RSPM (R 4.2.0)
##  MASS       * 7.3-57     2022-04-22 [2] RSPM (R 4.2.0)
##  Matrix     * 1.4-1      2022-03-23 [3] CRAN (R 4.2.0)
##  mice       * 3.14.0     2021-11-24 [2] RSPM (R 4.2.0)
##  NNaccess   * 0.2.1      2023-11-07 [1] RSPM (R 4.2.0)
##  NNadam     * 0.0.8      2023-08-18 [1] RSPM (R 4.2.0)
##  NNBiostat  * 4.2.0-40   2023-09-25 [2] RSPM (R 4.2.0)
##  NNcalc     * 1.0.5      2023-08-18 [1] RSPM (R 4.2.0)
##  NNexport   * 0.2.2      2023-08-18 [1] RSPM (R 4.2.0)
##  NNlog      * 0.2.1      2024-02-12 [1] RSPM (R 4.2.0)
##  NNplot     * 0.0.9      2023-08-18 [1] RSPM (R 4.2.0)
##  NNR        * 1.9.11     2023-09-25 [2] RSPM (R 4.2.0)
##  NNremote   * 0.3.3      2023-11-02 [1] RSPM (R 4.2.0)
```

R version

OS

Packages

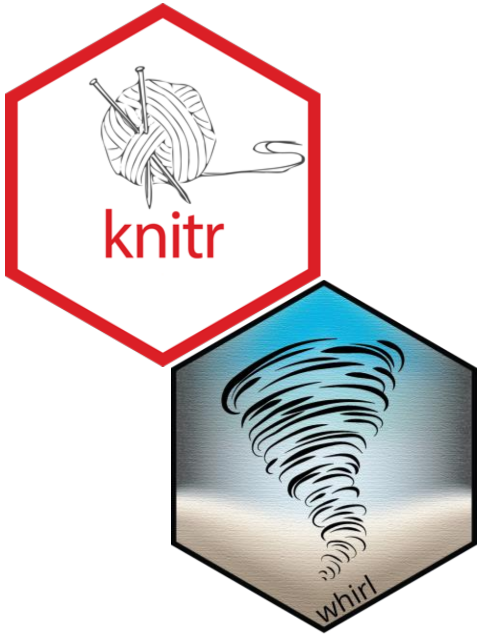


```
##  NNadam     * 0.0.8      2023-08-18 [1] RSPM (R 4.2.0)
##  NNBiostat  * 4.2.0-40   2023-09-25 [2] RSPM (R 4.2.0)
##  NNcalc     * 1.0.5      2023-08-18 [1] RSPM (R 4.2.0)
##  NNexport   * 0.2.2      2023-08-18 [1] RSPM (R 4.2.0)
##  NNlog      * 0.2.1      2024-02-12 [1] RSPM (R 4.2.0)
##  NNplot     * 0.0.9      2023-08-18 [1] RSPM (R 4.2.0)
##  NNR        * 1.9.11     2023-09-25 [2] RSPM (R 4.2.0)
##  NNremote   * 0.3.3      2023-11-02 [1] RSPM (R 4.2.0)
##  NNrender   * 0.2.10     2023-08-18 [1] RSPM (R 4.2.0)
##  NNsdtm     * 0.0.1      2023-09-25 [2] RSPM (R 4.2.0)
##  NNtable    * 0.1.5      2023-08-18 [1] RSPM (R 4.2.0)
##  NNtfl      * 0.1.4.9004 2023-11-10 [1] RSPM (R 4.2.0)
##  NNutility  * 0.1.1      2023-08-18 [1] RSPM (R 4.2.0)
##  patchwork  * 1.1.1      2020-12-17 [2] RSPM (R 4.2.0)
##  purrr      * 0.3.4      2020-04-17 [2] RSPM (R 4.2.0)
##  readr      * 2.1.2      2022-01-30 [2] RSPM (R 4.2.0)
##  stringr    * 1.5.1      2023-11-14 [1] RSPM (R 4.2.0)
##  survival   * 3.3-1      2022-03-03 [3] CRAN (R 4.2.0)
##  tibble     * 3.1.7      2022-05-03 [2] RSPM (R 4.2.0)
##  tidyr      * 1.2.0      2022-02-01 [2] RSPM (R 4.2.0)
##
##  [1] /scer/homedirs/ktqn/R/x86_64-pc-linux-gnu-library/4.2
##  [2] /opt/R/4.2.0/lib/R/site-library
##  [3] /opt/R/4.2.0/lib/R/library
```

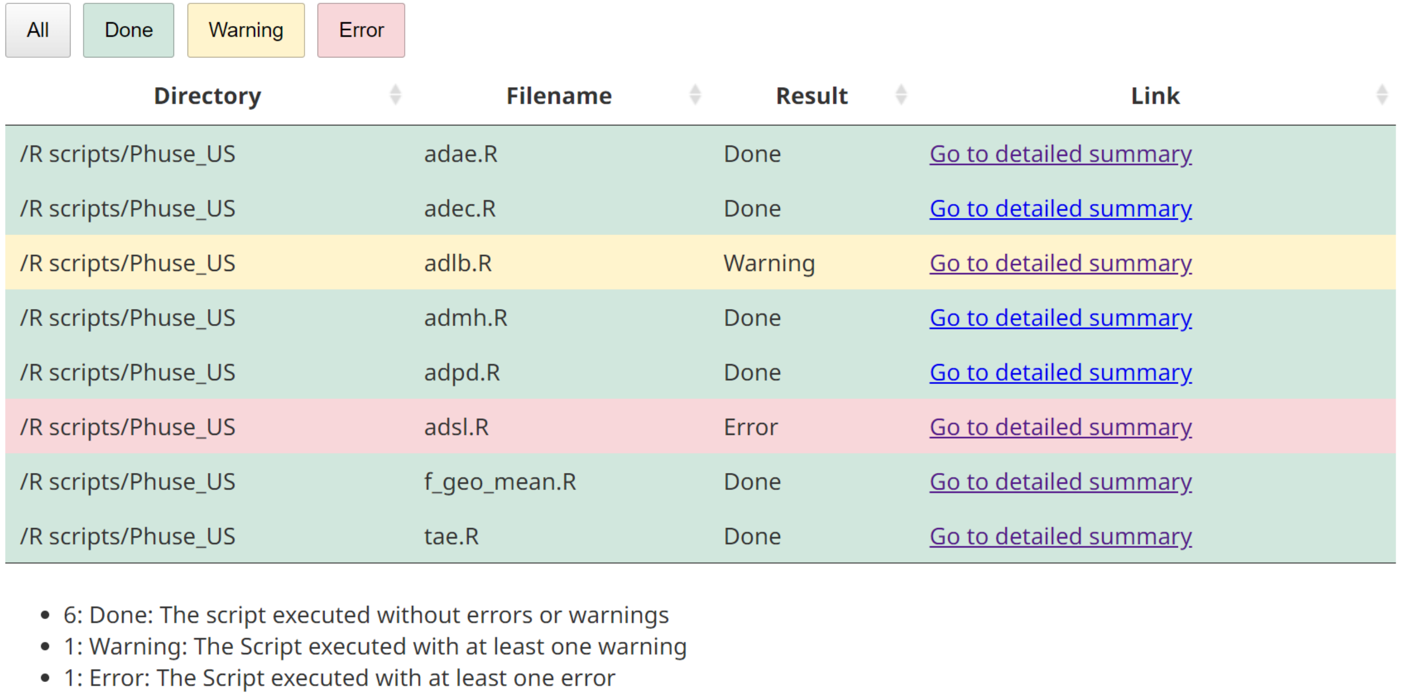
▶ The versions of packages loaded via a namespace (and not attached):

▶ Information on all packages available in the libraries:

▶ Options set for the computing environment:



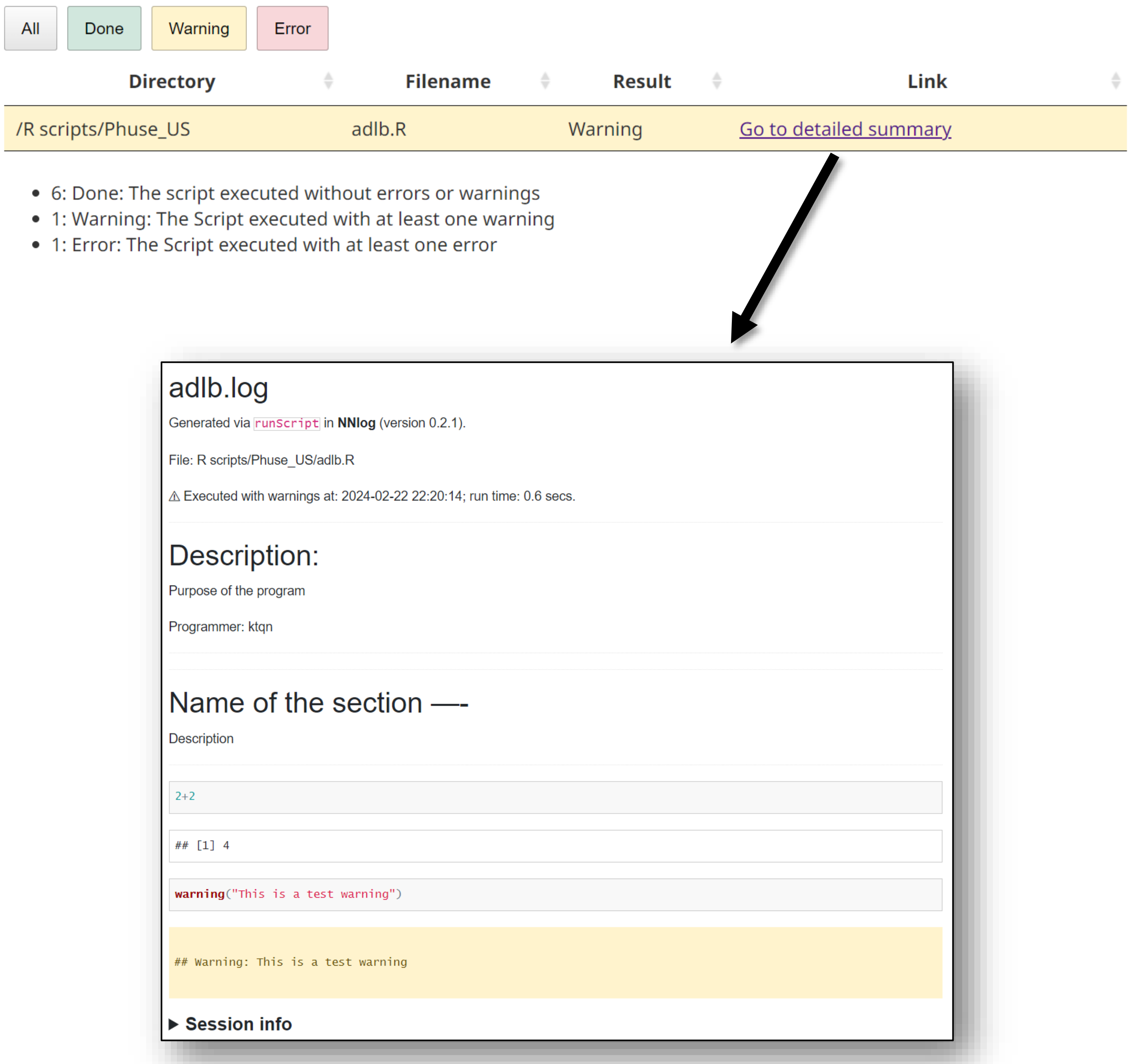
Running multiple programs



Directory	Filename	Result	Link
/R scripts/Phuse_US	adae.R	Done	Go to detailed summary
/R scripts/Phuse_US	adec.R	Done	Go to detailed summary
/R scripts/Phuse_US	adlb.R	Warning	Go to detailed summary
/R scripts/Phuse_US	admh.R	Done	Go to detailed summary
/R scripts/Phuse_US	adpd.R	Done	Go to detailed summary
/R scripts/Phuse_US	adsl.R	Error	Go to detailed summary
/R scripts/Phuse_US	f_geo_mean.R	Done	Go to detailed summary
/R scripts/Phuse_US	tae.R	Done	Go to detailed summary

- 6: Done: The script executed without errors or warnings
- 1: Warning: The Script executed with at least one warning
- 1: Error: The Script executed with at least one error

Running multiple programs



Directory	Filename	Result	Link
/R scripts/Phuse_US	adlb.R	Warning	Go to detailed summary

- 6: Done: The script executed without errors or warnings
- 1: Warning: The Script executed with at least one warning
- 1: Error: The Script executed with at least one error

adlb.log

Generated via `runScript` in **NNlog** (version 0.2.1).

File: R scripts/Phuse_US/adlb.R

⚠ Executed with warnings at: 2024-02-22 22:20:14; run time: 0.6 secs.

Description:

Purpose of the program

Programmer: ktqn

Name of the section —-

Description

```
2+2
```

```
## [1] 4
```

```
warning("This is a test warning")
```

```
## Warning: This is a test warning
```

▶ **Session info**

The three essentials

Accuracy How do we ensure programs are accurate?

Traceability How do we document what we have done?

Reproducibility How to ensure reproducibility when using R

The three essentials

Accuracy	**How do we ensure programs are accurate?**
Traceability	**How do we document what we have done?**
Reproducibility	**How to ensure reproducibility when using R** • Package synchronization within a project • Support authorities in environment replication

The essentials: Reproducibility

Internal distribution

posit Package Manager

- Manage R packages
- Sharing external packages
- Share internal packages
- Create subsets of approved packages

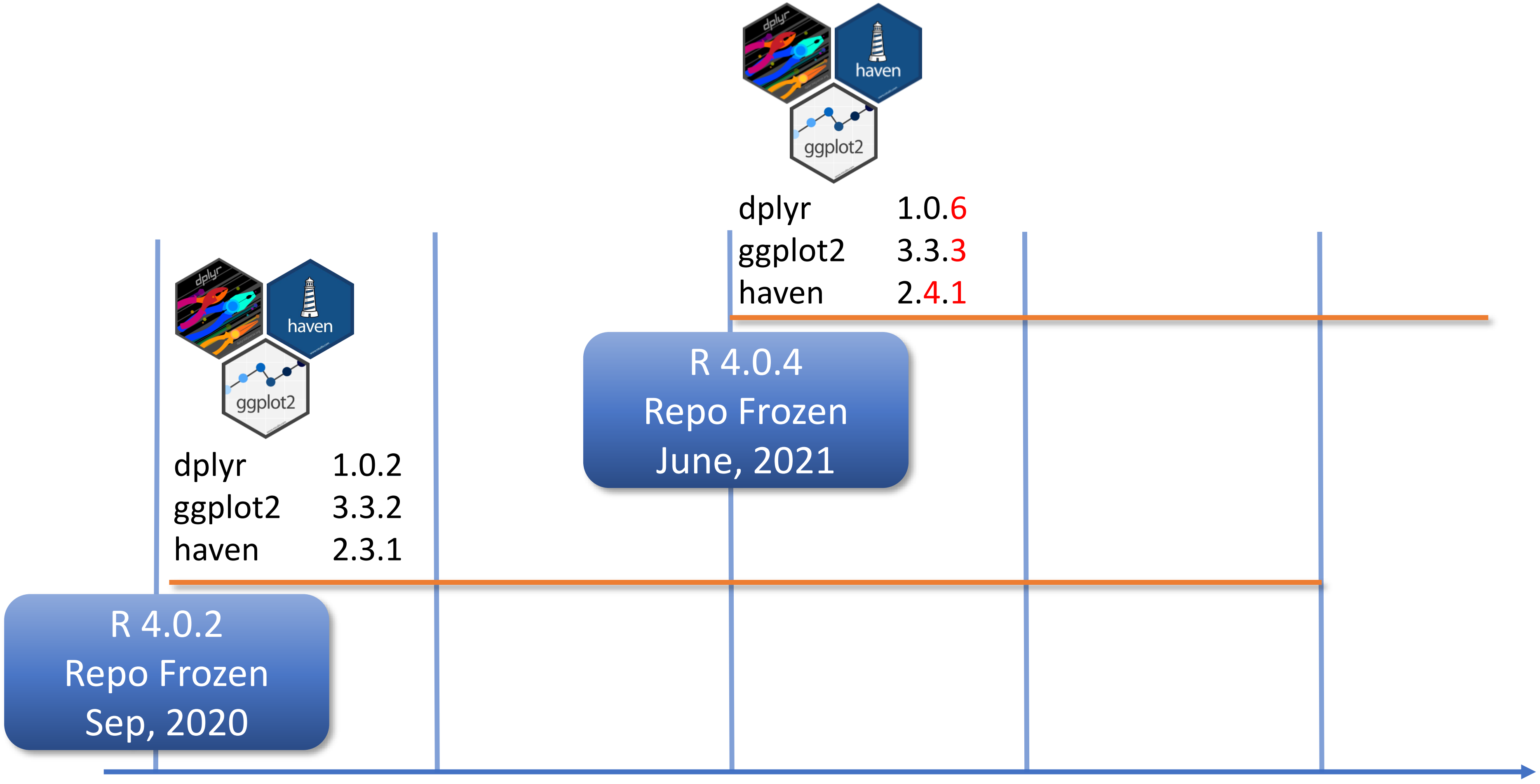
Synchronization

renv

External distribution

pkglite

The shared-baseline model



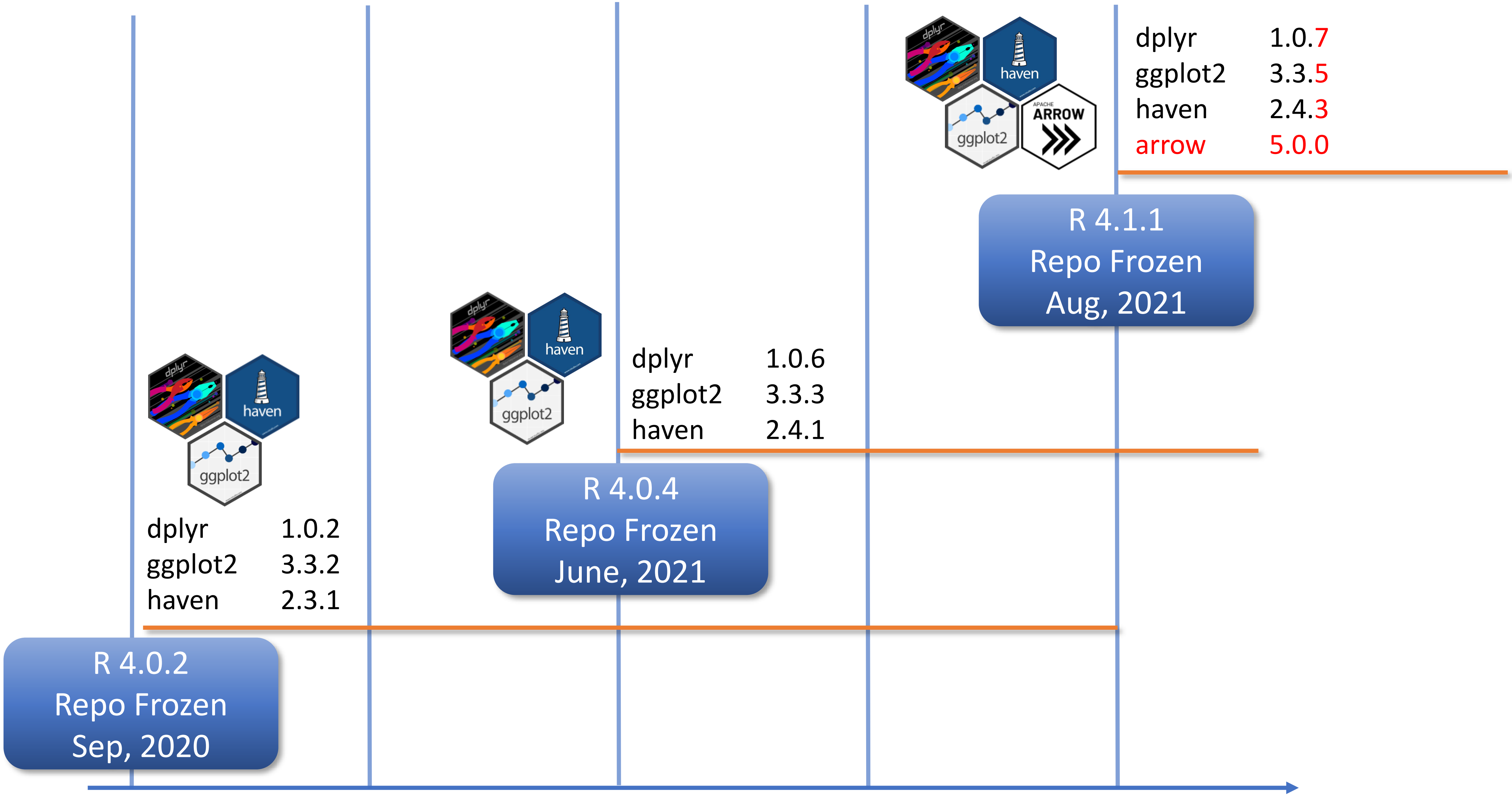
R 4.0.2
Repo Frozen
Sep, 2020

dplyr	1.0.2
ggplot2	3.3.2
haven	2.3.1

R 4.0.4
Repo Frozen
June, 2021

dplyr	1.0.6
ggplot2	3.3.3
haven	2.4.1

The shared-baseline model



R 4.0.2
Repo Frozen
Sep, 2020

dplyr	1.0.2
ggplot2	3.3.2
haven	2.3.1

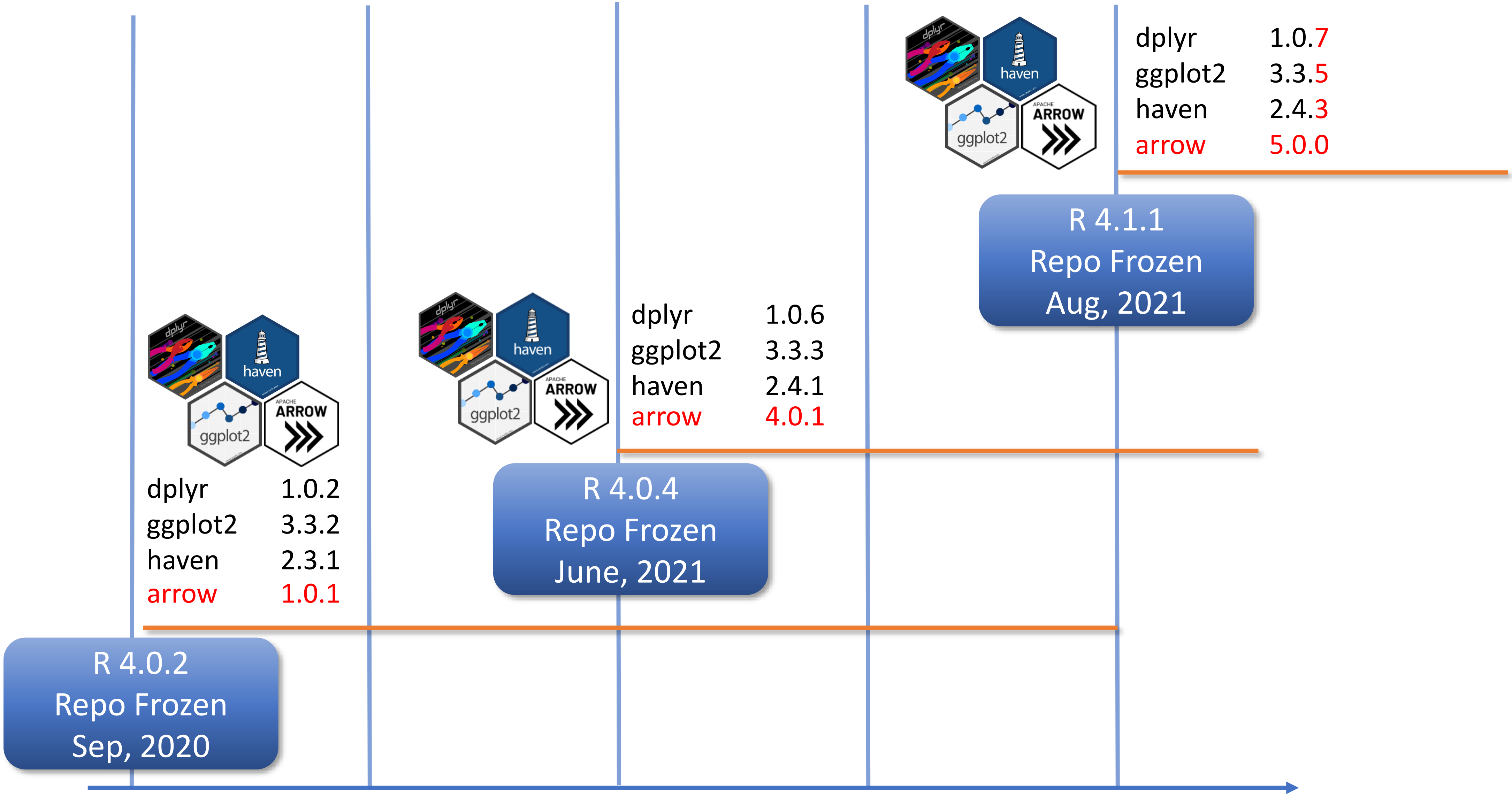
R 4.0.4
Repo Frozen
June, 2021

dplyr	1.0.6
ggplot2	3.3.3
haven	2.4.1

R 4.1.1
Repo Frozen
Aug, 2021

dplyr	1.0.7
ggplot2	3.3.5
haven	2.4.3
arrow	5.0.0

The shared-baseline model



R 4.0.2
Repo Frozen
Sep, 2020

dplyr	1.0.2
ggplot2	3.3.2
haven	2.3.1
arrow	1.0.1

R 4.0.4
Repo Frozen
June, 2021

dplyr	1.0.6
ggplot2	3.3.3
haven	2.4.1
arrow	4.0.1

R 4.1.1
Repo Frozen
Aug, 2021

dplyr	1.0.7
ggplot2	3.3.5
haven	2.4.3
arrow	5.0.0

Internal packages

- Several internal packages to streamline output generation

- Enable fast turn-around
  - Not locked to a specific R version
  - Shared-baseline + {renv} to keep track of these

The essentials: Reproducibility

Internal distribution

posit Package Manager

- Manage R packages
- Sharing external packages
- Share internal packages
- Create subsets of approved packages

Synchronization

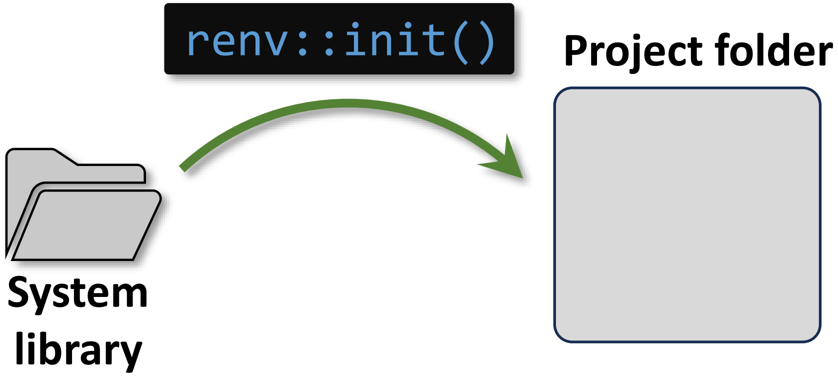
renv

- Project-wise synchronization of internal & external packages
- Enable reproducing the same environment in another place

External distribution

pkglite

The essentials: Reproducibility



```
renv::init()
```

```
renv::init()
```

1) Creates renv/activate.R

2) Update .Rprofile with:
   source("renv/activate.R")

3) Creates renv/library (project library)

4) Initiate the lockfile → renv.lock

```
├── project.Rproj
```

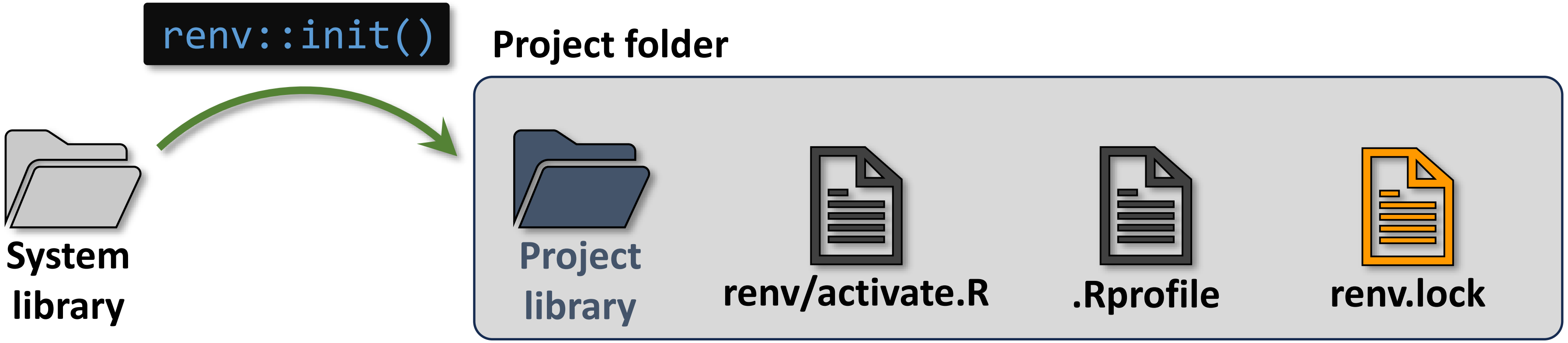
renv/activate.R

.Rprofile

Project library

renv.lock

The essentials: Reproducibility



```
renv::init()
```

Project folder

System library

Project library

renv/activate.R

.Rprofile

renv.lock

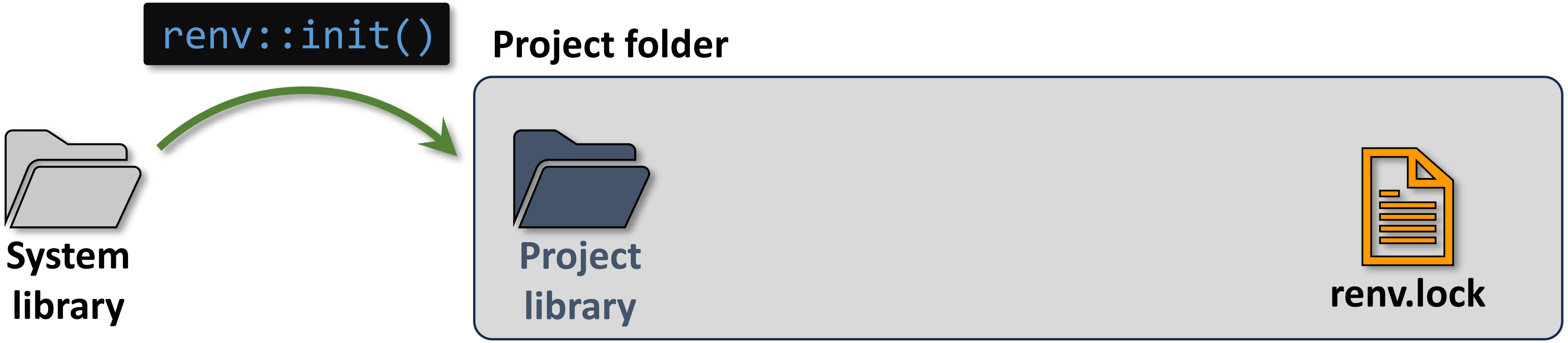
```
{
 "R": {
  "Version": "4.3.1",
  "Repositories": [
   {
    "Name": "CRAN",
    "URL": "https://rspm.bifrost-dev.corp.aws.novonordisk.com/default-cran/2023-10-25"
   },
   {
    "Name": "NN Prod",
    "URL": "https://rspm.bifrost-dev.corp.aws.novonordisk.com/prod-internal-4.3.1/latest"
   }
  ]
 },
 "Packages": {
  "renv": {
   "Package": "renv",
   "Version": "1.0.3",
   "Source": "Repository",
   "Repository": "RSPM",
   "Requirements": [
    "utils"
   ],
   "Hash": "41b847654f567341725473431dd0d5ab“
  }
 }
}
```

Private package manager for **CRAN** packages

Private package manager for **internal** packages

renv.lock

The essentials: Reproducibility



`renv::init()`

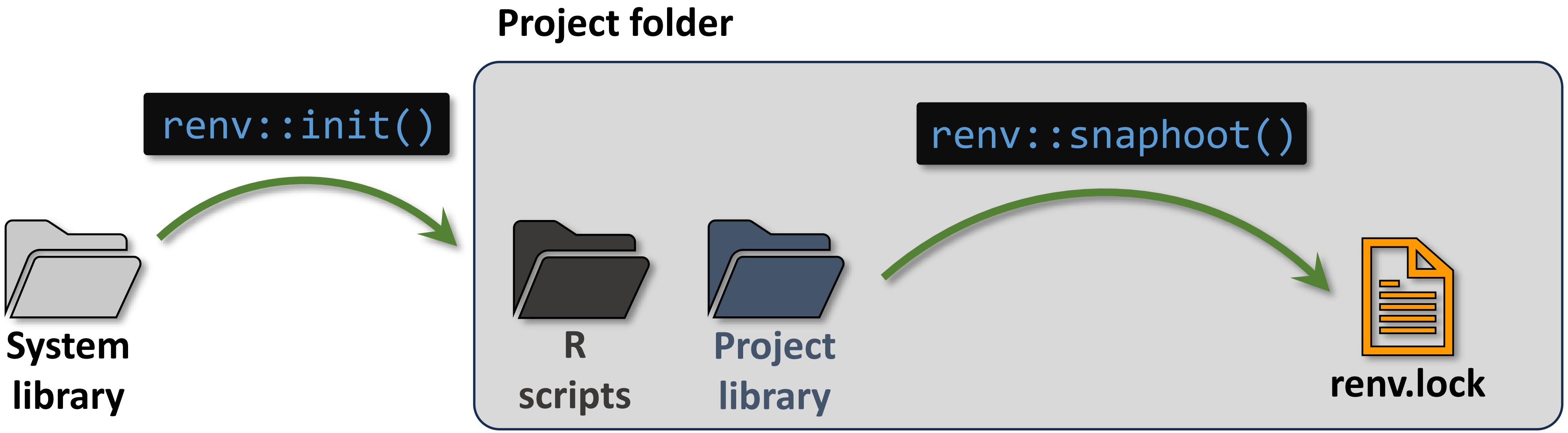
System library

Project folder

Project library

renv.lock

The essentials: Reproducibility



Project folder

```
renv::init()
```

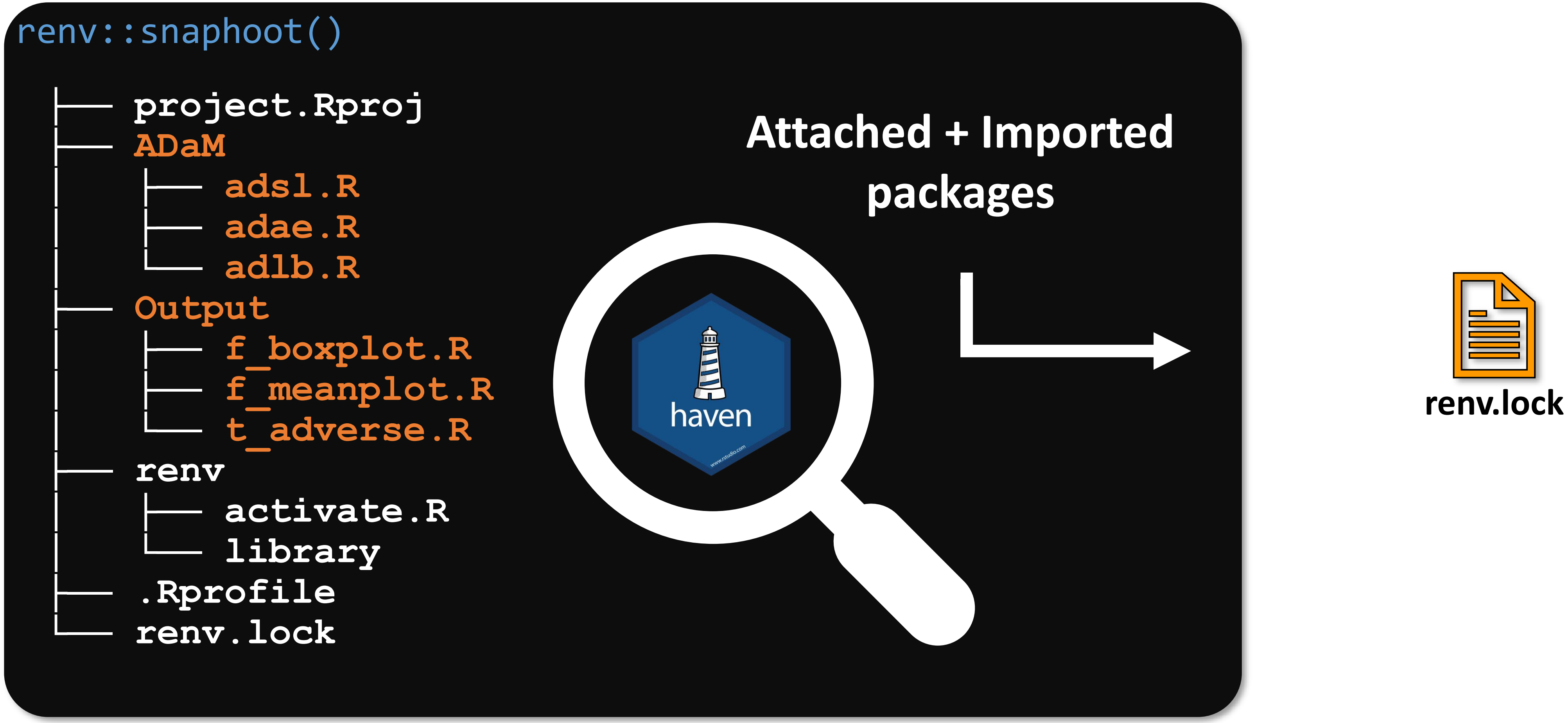
```
renv::snaphoot()
```

System library

R scripts

Project library

renv.lock



```
renv::snaphoot()

├── project.Rproj
├── ADaM
│   ├── adsl.R
│   ├── adae.R
│   └── adlb.R
├── Output
│   ├── f_boxplot.R
│   ├── f_meanplot.R
│   └── t_adverse.R
├── renv
│   ├── activate.R
│   └── library
├── .Rprofile
└── renv.lock
```

Attached + Imported packages

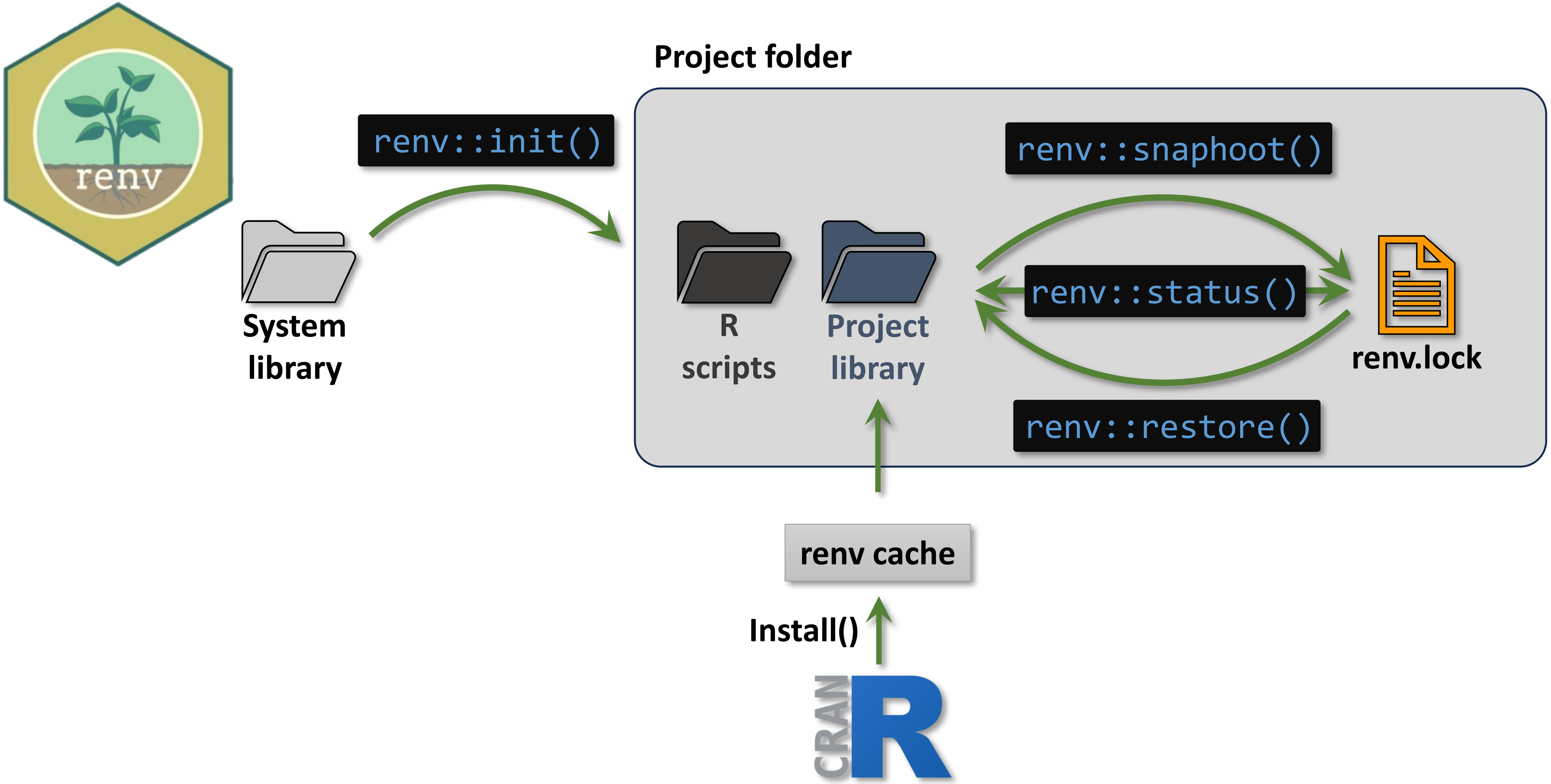
```
{
 "R": {
  "Version": "4.3.1",
  "Repositories": [
   {
    "Name": "CRAN",
    "URL": "https://rspm.bifrost-dev.corp.aws.novonordisk.com/default-cran/2023-10-25"
   },
   {
    "Name": "NN Prod",
    "URL": "https://rspm.bifrost-dev.corp.aws.novonordisk.com/prod-internal-4.3.1/latest"
   }
  ]
 },
 "Packages": {
  "cli": {
   "Package": "cli",
   "Version": "3.6.1",
   "Source": "Repository",
   "Repository": "RSPM",
   "Requirements": [
    "R",
    "utils"
   ],
   "Hash": "89e6d8219950eac806ae0c489052048a"
  },
  "clipr": {
   "Package": "clipr",
```

renv.lock

```
},
"haven": {
 "Package": "haven",
 "Version": "2.5.3",
 "Source": "Repository",
 "Repository": "RSPM",
 "Requirements": [
  "R",
  "cli",
  "cpp11",
  "forcats",
  "hms",
  "lifecycle",
  "methods",
  "readr",
  "rlang",
  "tibble",
  "tidyselect",
  "vctrs"
 ],
 "Hash": "9b302fe352f9cfc5dcf0a4139af3a565"
},
"hms": {
 "Package": "hms",
 "Version": "1.1.3",
 "Source": "Repository",
 "Repository": "RSPM",
 "Requirements": [
```

renv.lock

The essentials: Reproducibility



renv

Project folder

renv::init()

renv::snaphoot()

renv::status()

renv::restore()

System library

R scripts

Project library

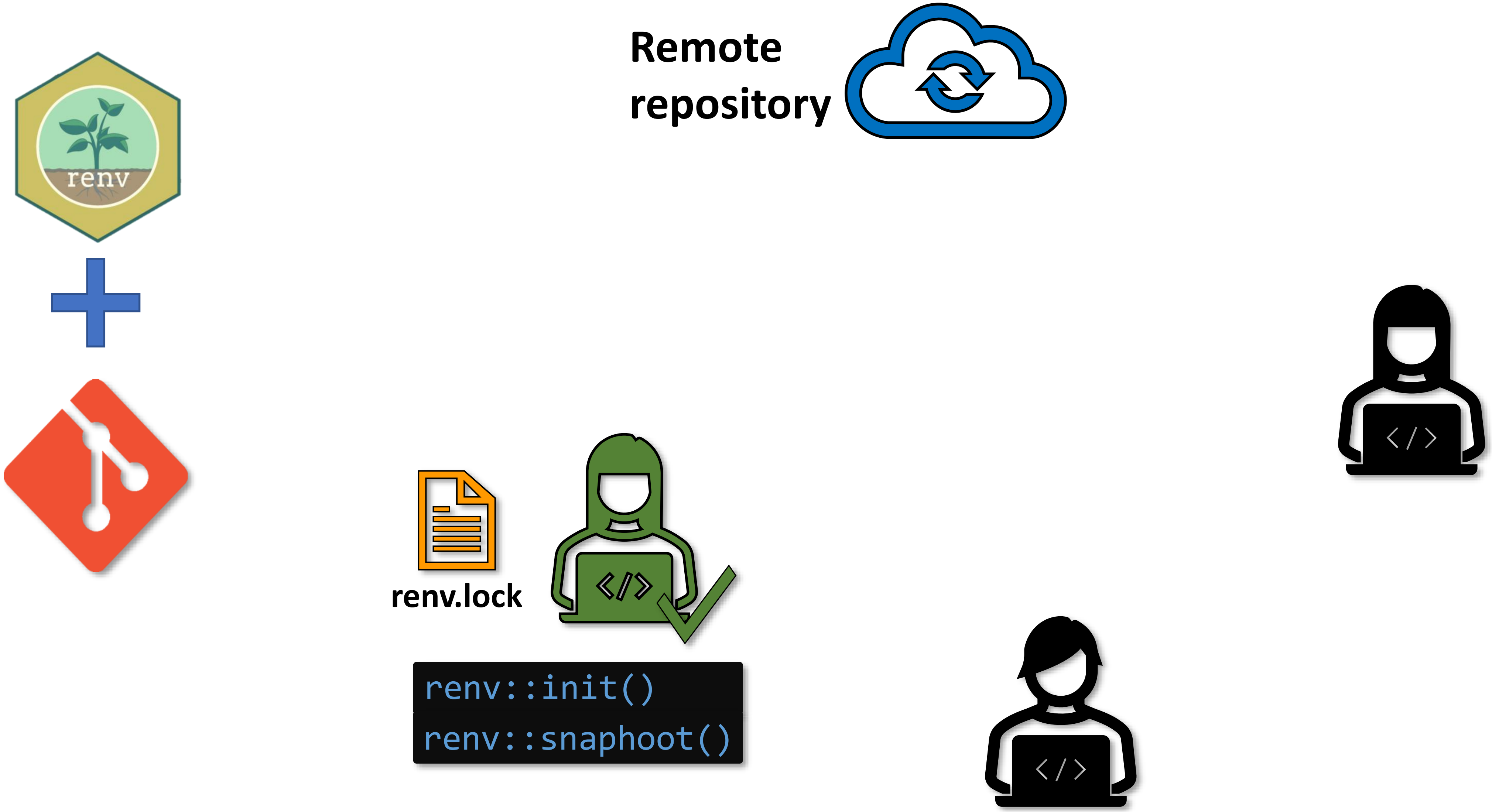
renv.lock

renv cache

Install()

CRAN R

The essentials: Reproducibility

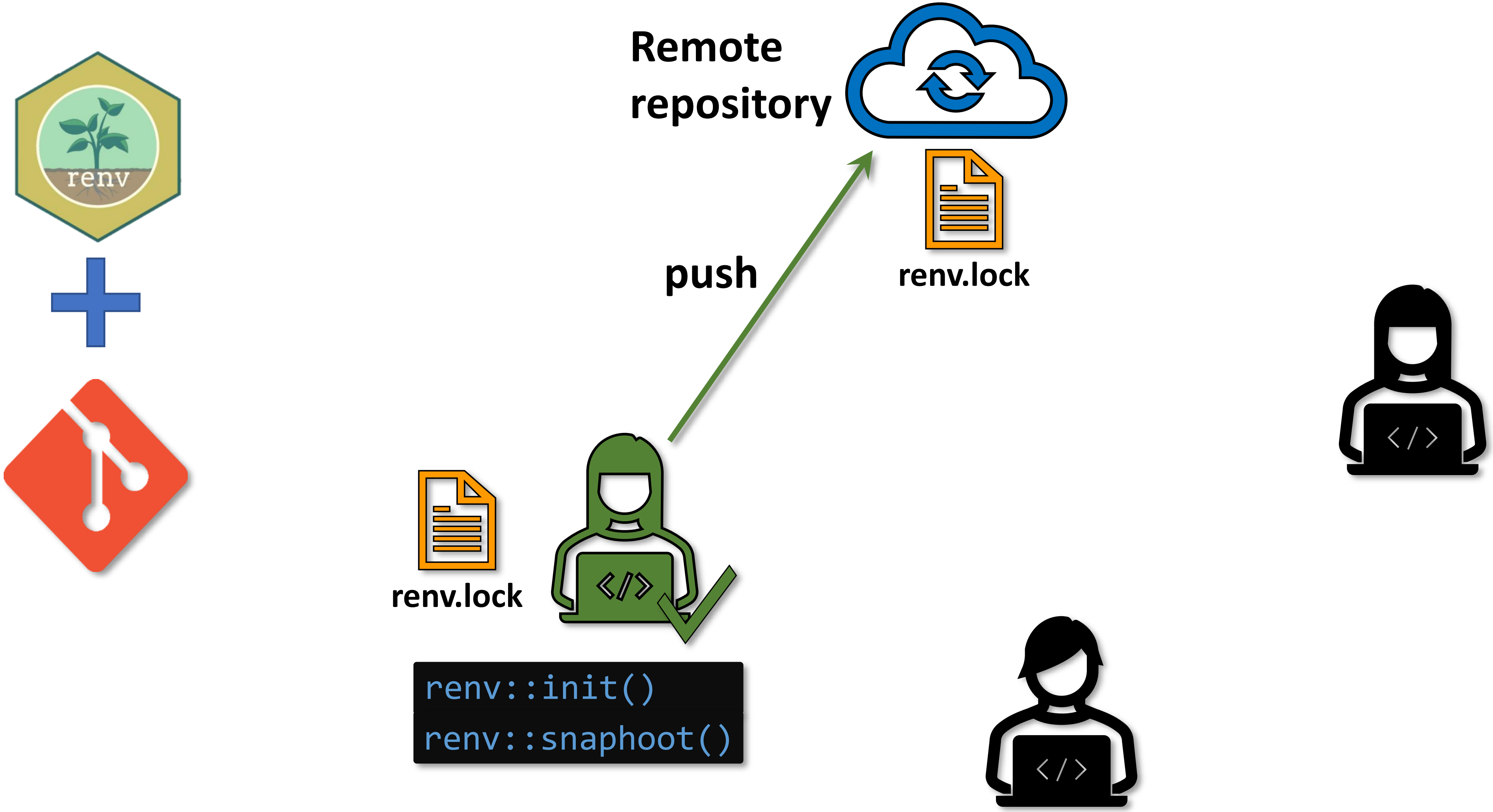


Remote repository

renv.lock

```
renv::init()
renv::snaphoot()
```

The essentials: Reproducibility



Remote repository

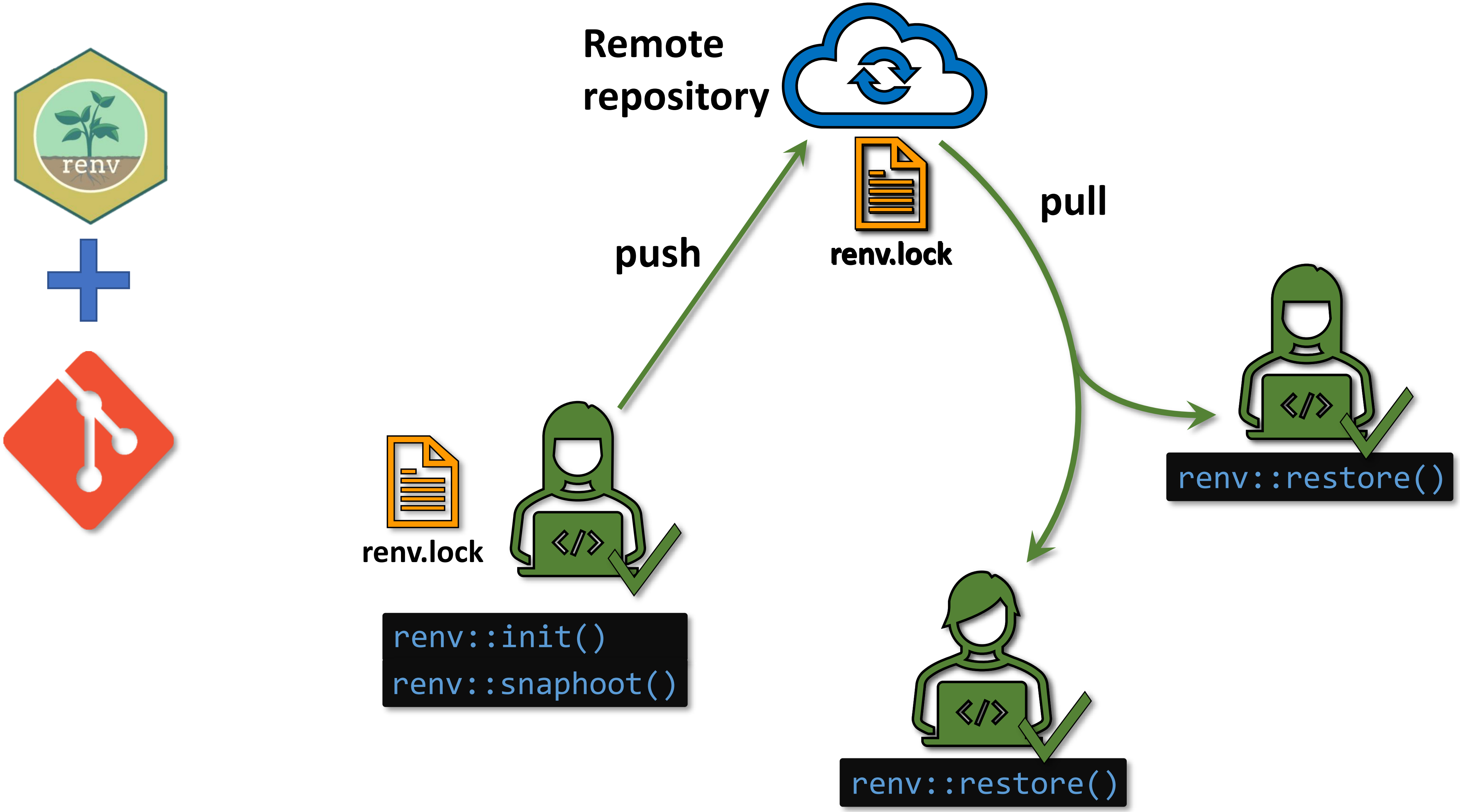
push

renv.lock

renv.lock

```
renv::init()
renv::snaphoot()
```

The essentials: Reproducibility

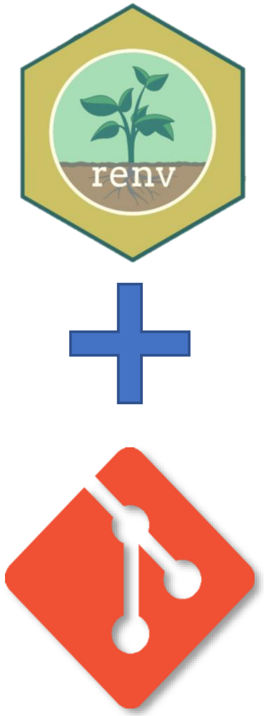


```
renv::init()
renv::snaphoot()
```

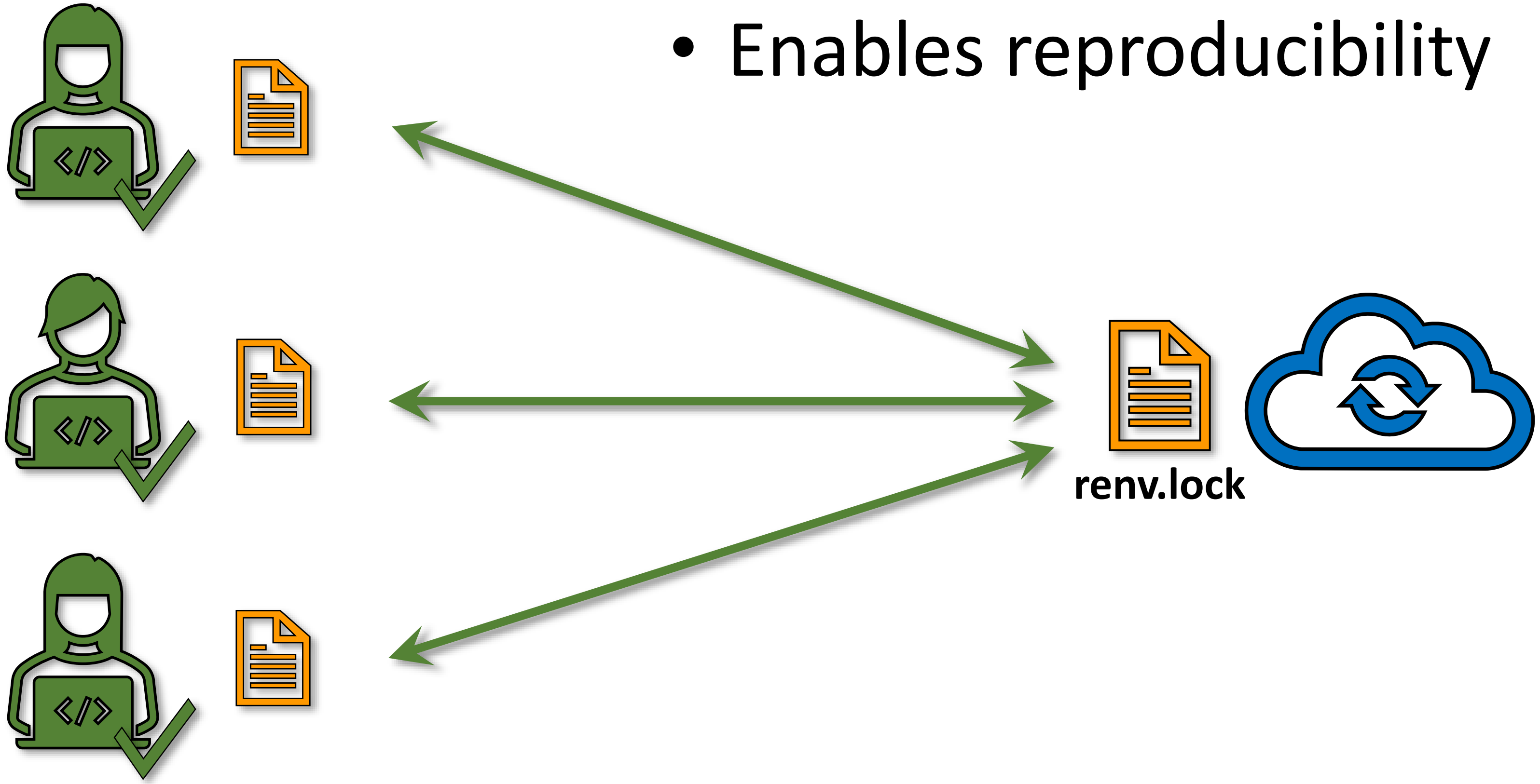
```
renv::restore()
```

```
renv::restore()
```

The essentials: Reproducibility



- Ensures that everyone use the same packages
- Enables reproducibility



The essentials: Reproducibility

Internal distribution

posit Package Manager

- Manage R packages
- Sharing external packages
- Share internal packages
- Create subsets of approved packages

Synchronization

renv

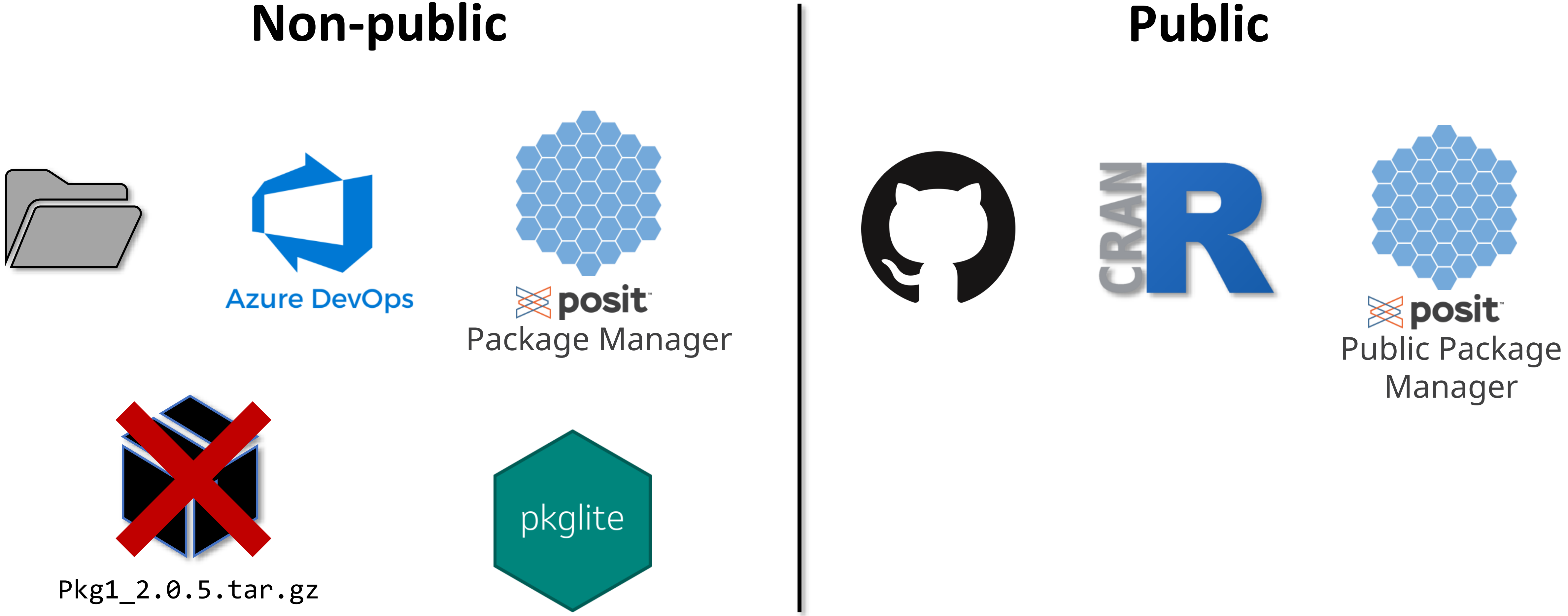
- Project-wise synchronization of internal & external packages
- Enable reproducing the same environment in another place

External distribution

pkglite

- Way to share internally developed packages as .txt files
- Enable distribution to authorities

Sharing packages with authorities



Non-public

Azure DevOps

posit
Package Manager

Pkg1_2.0.5.tar.gz

pkglite

Public

CRAN R

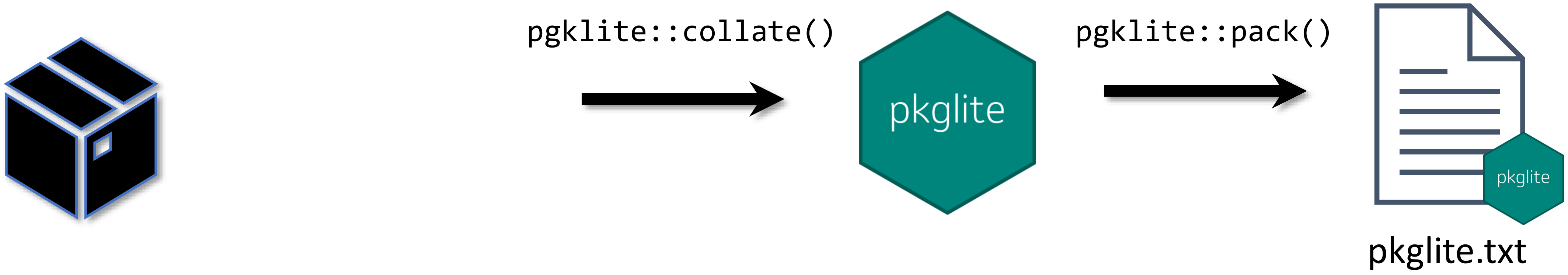
posit
Public Package Manager

Pack packages

```
├── DESCRIPTION
├── inst
│   └── extdata
├── man
│   └── *.Rd
├── NAMESPACE
├── NEWS.md
├── R
│   └── *.R
├── README.md
├── tests
│   ├── testthat
│   │   └── *.R
│   └── testthat.R
└── vignettes
    └── *.Rmd
```

pkglite

Pack packages



pgklite::collate()

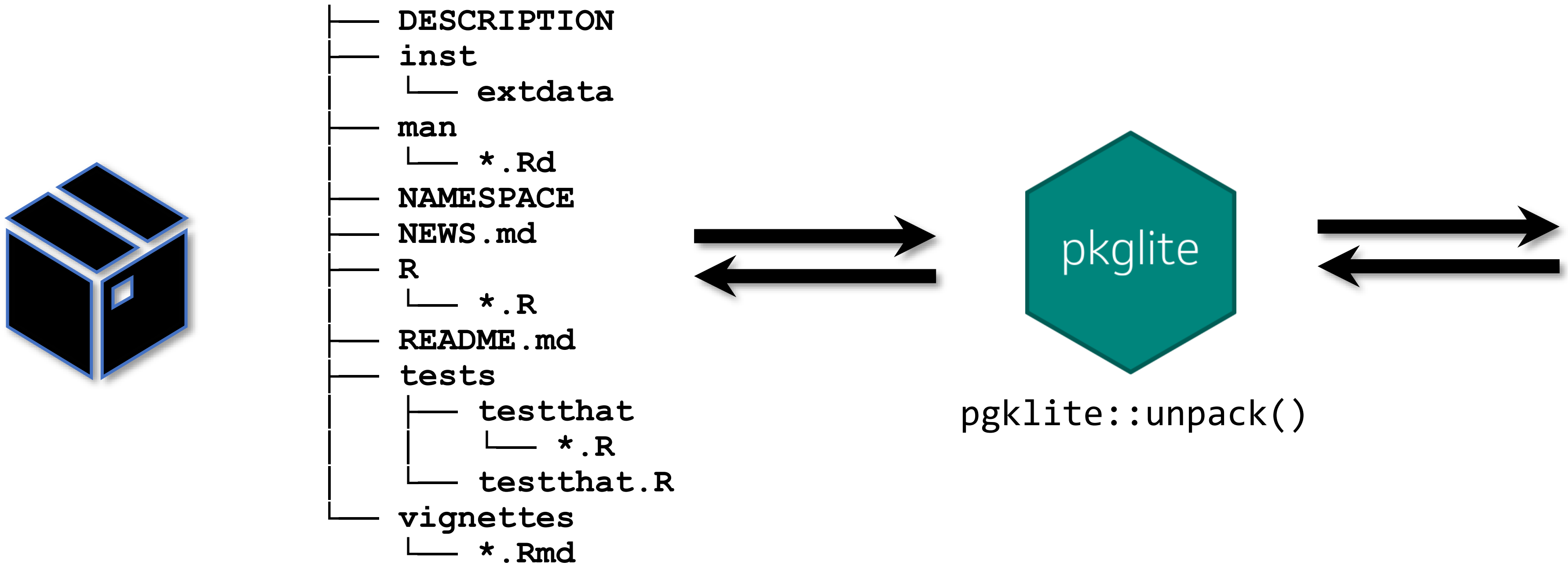
pkglite

pgklite::pack()

pkglite

pkglite.txt

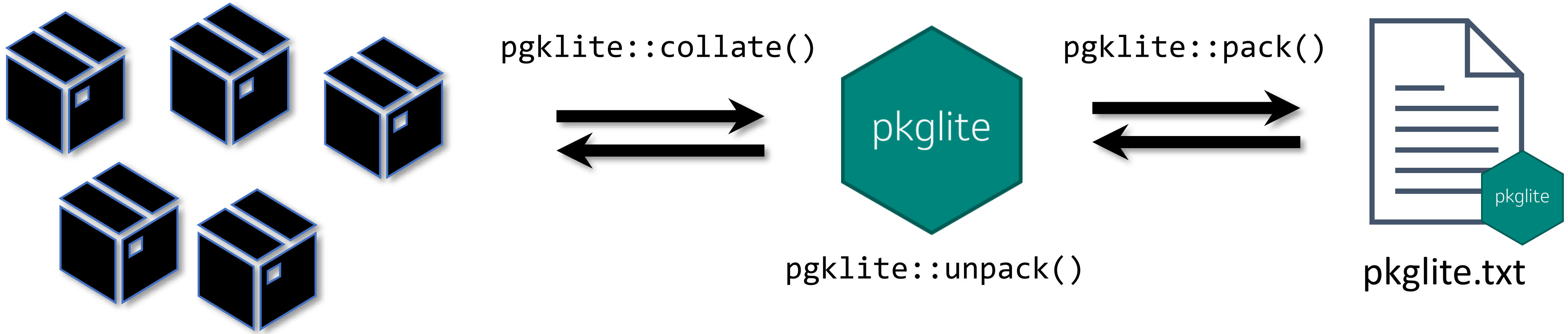
Pack packages



```
├── DESCRIPTION
├── inst
│   └── extdata
├── man
│   └── *.Rd
├── NAMESPACE
├── NEWS.md
├── R
│   └── *.R
├── README.md
├── tests
│   ├── testthat
│   │   └── *.R
│   └── testthat.R
└── vignettes
    └── *.Rmd
```

pgklite::unpack()

Pack packages



pgklite::collate()

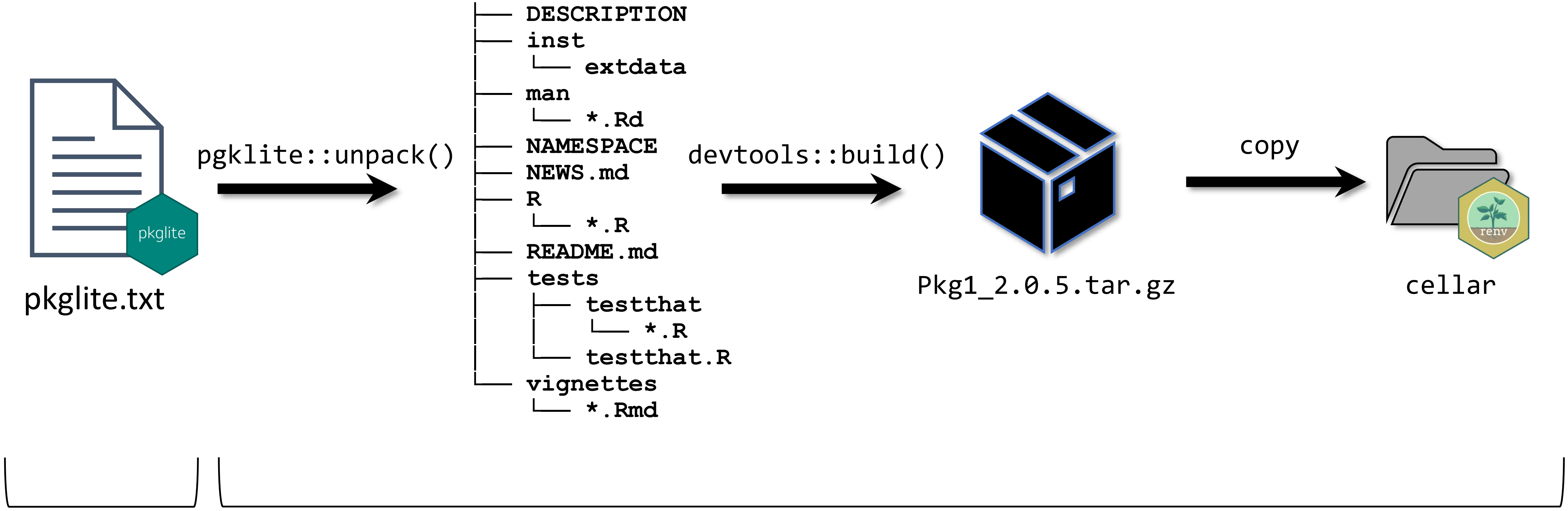
pgklite::pack()

pkglite

pgklite::unpack()

pkglite.txt

Combine pgklite with renv



```
├── DESCRIPTION
├── inst
│   └── extdata
├── man
│   └── *.Rd
├── NAMESPACE
├── NEWS.md
├── R
│   └── *.R
├── README.md
├── tests
│   ├── testthat
│   │   └── *.R
│   └── testthat.R
└── vignettes
    └── *.Rmd
```

Send to authorities

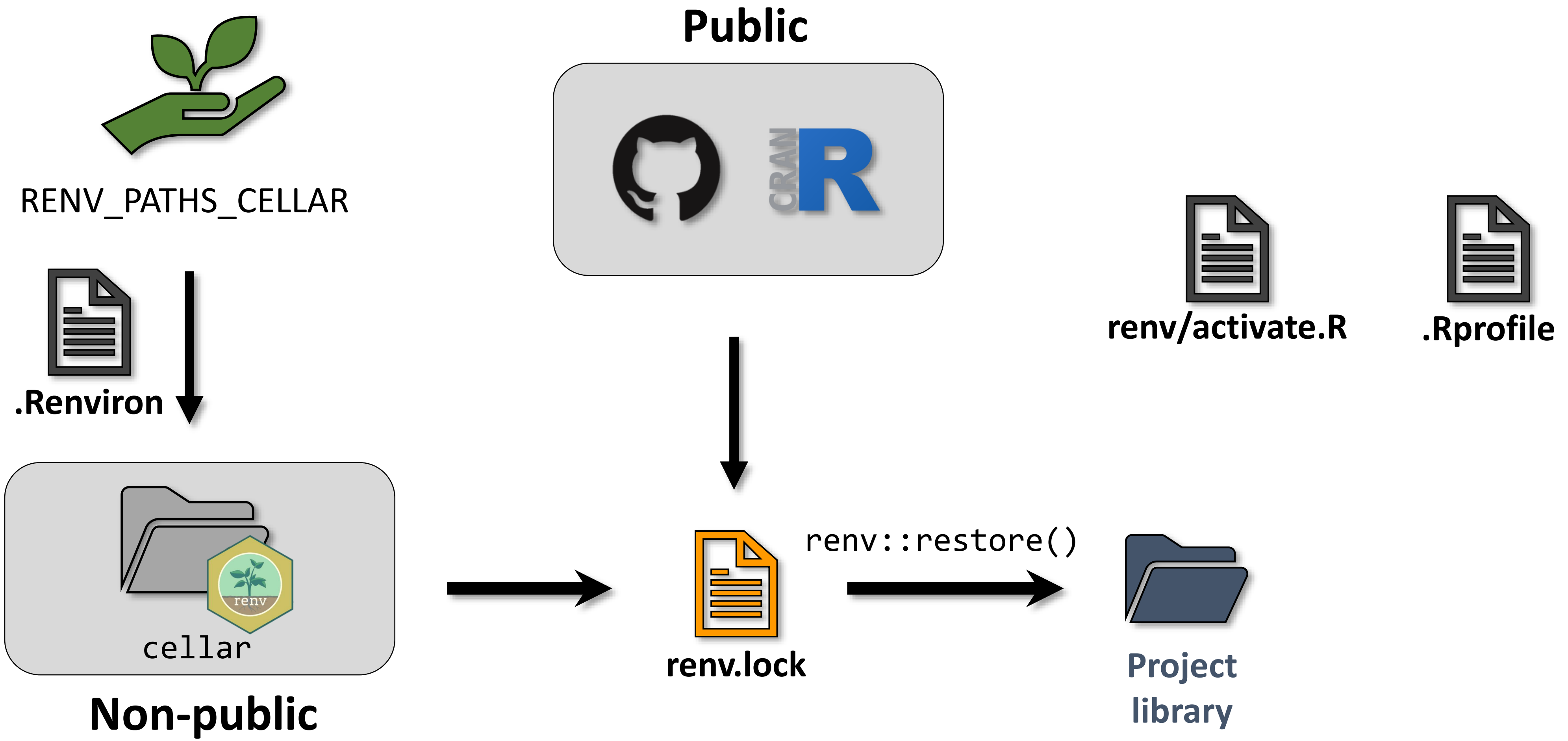
Happens at the receiver

Combine pgklite with renv

cellar

```
├── pkg1
│   ├── pkg1_2.0.0.tar.gz
│   └── pkg1_2.0.5.tar.gz
├── pkg2
│   └── pkg2_1.1.1.tar.gz
├── pkg3
│   ├── pkg3_4.1.0.tar.gz
│   ├── pkg3_4.5.0.tar.gz
│   ├── pkg3_5.1.0.tar.gz
│   └── pkg3_5.2.1.tar.gz
└── pkg4
    └── pkg4_2.1.1.tar.gz
```

Combine pgklite with renv



Public

RENV_PATHS_CELLAR

.Renviron

renv/activate.R

.Rprofile

cellar

Non-public

renv.lock

`renv::restore()`

Project library

Reproducibility

Exercise

Exercise

Background:

- We have submitted several trials to FDA
- Slightly different versions of our internal packages for each trial
- Same external packages on all trials
- We have used {renv} on all trials
  → **renv.lock** exist on each trial

Authorities asked for:

- Specific analysis using {Rstat} - a package not currently captured in renv.lock
- Reproducibility (they want to execute the programs)

End product at recipient:

```
├── project.Rproj
├── SDTM data
│   ├── ae.xpt
│   └── lb.xpt
├── ADaM data
│   ├── adsl.xpt
│   ├── adae.xpt
│   └── adlb.xpt
├── ADaM programs
│   ├── adsl.R
│   ├── adae.R
│   └── adlb.R
├── Output programs
│   ├── a_new_analysis.R
│   ├── f_meanplot.R
│   └── t_adverse.R
├── Output
├── renv
│   ├── activate.R
│   └── library
├── .Rprofile
└── renv.lock
```

Exercise

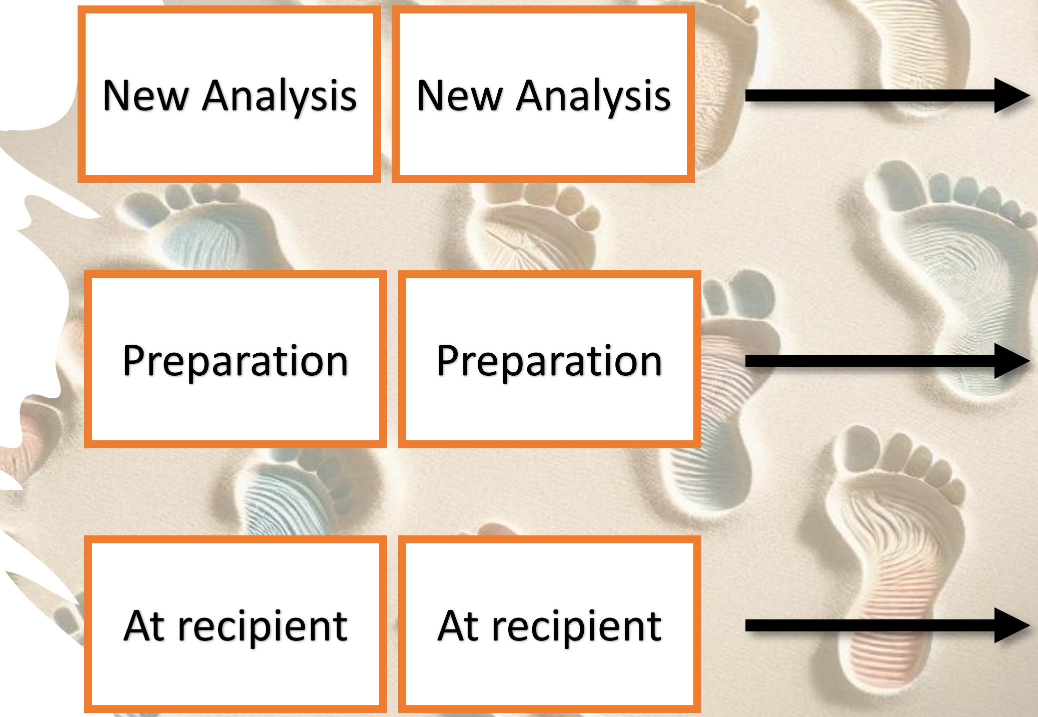
Background:

- We have submitted several trials to FDA
- Slightly different versions of our internal packages for each trial
- Same external packages on all trials
- We have used {renv} on all trials
  → **renv.lock** exist on each trial

Authorities asked for:

- Specific analysis using {Rstat} - a package not currently captured in renv.lock
- Reproducibility (they want to execute the programs)

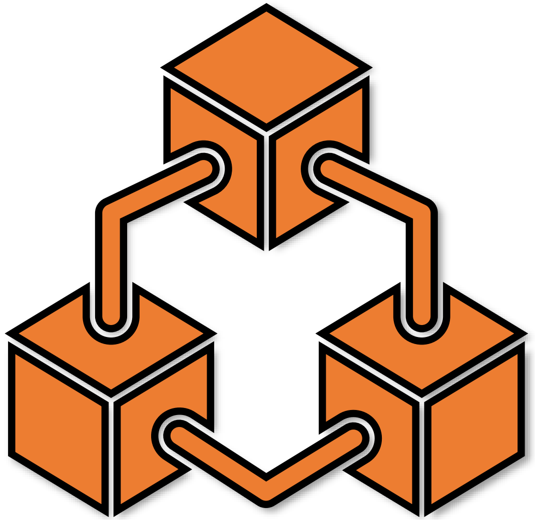
Three groups of tasks
Each group should be arranged in sequence



New Analysis | New Analysis →

Preparation | Preparation →

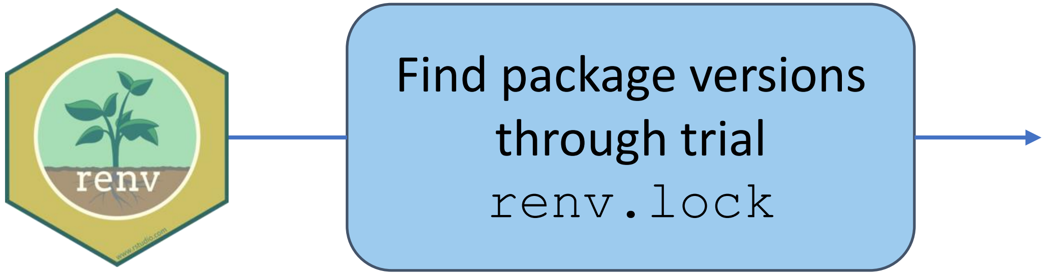
At recipient | At recipient →



Reproducibility

Summary
&
Example from Novo Nordisk

Packing Packages



Find package versions through trial

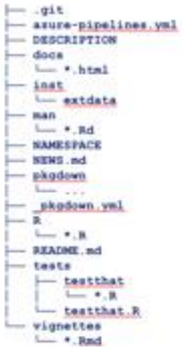
`renv.lock`

Packing Packages

pharmaverse

renv

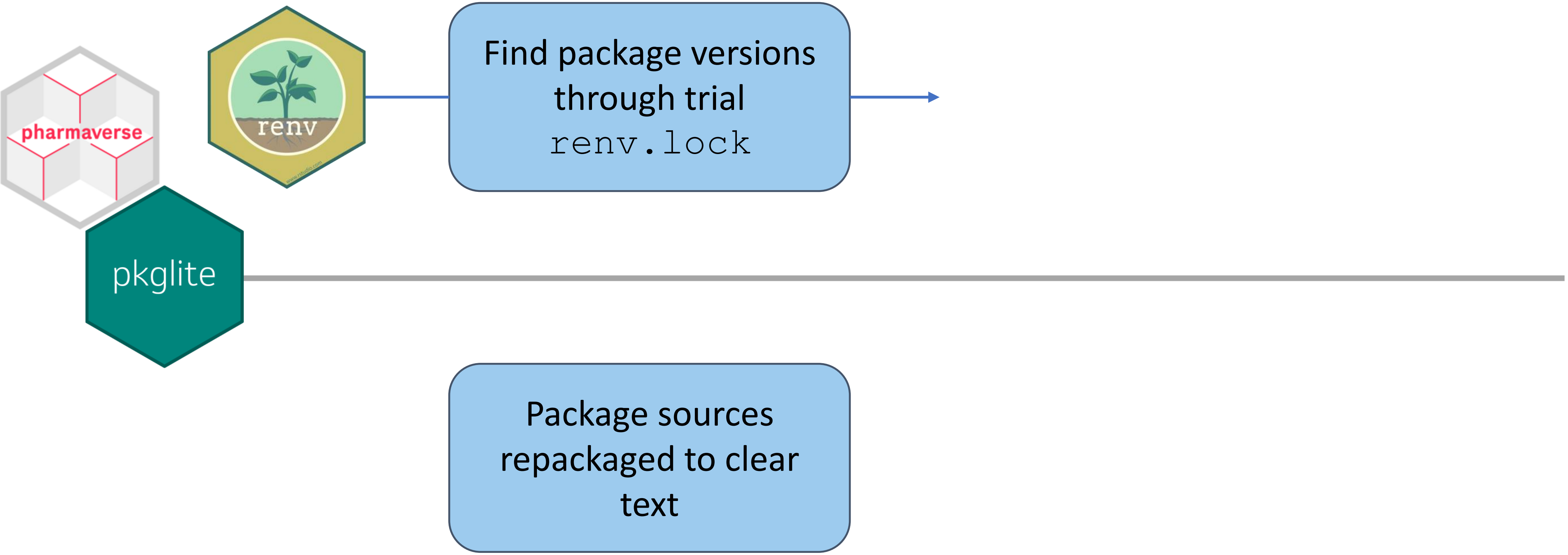
Find package versions through trial
`renv.lock`



```
├── .git
├── azure-pipelines.yml
├── DESCRIPTION
├── docs
│   └── *.html
├── inst
│   └── extdata
├── man
│   └── *.Rd
├── NAMESPACE
├── NEWS.md
├── pkgdown
│   └── ...
├── _pkgdown.yml
├── R
│   └── *.R
├── README.md
├── tests
│   ├── testthat
│   │   └── *.R
│   └── testthat.R
└── vignettes
    └── *.Rmd
```

pkglite

Packing Packages



Find package versions through trial
`renv.lock`

Package sources repackaged to clear text

Send to authorities

pkglite-{trial}.txt

Re-packaged NN packages

renvlock-{trial}.txt

`renv.lock` with reference to MRAN

activate.txt

`renv` bootstrapping script

buildrenviron.r

Unpack, setup, rebuild and initialise environment

```
{
 "R": {
  "Version": "4.3.1",
  "Repositories": [
   {
    "Name": "CRAN",
    "URL": "https://rspm.bifrost-dev.corp.aws.novonordisk.com/default-cran/2023-10-25"
   },
   {
    "Name": "NN Prod",
    "URL": "https://rspm.bifrost-dev.corp.aws.novonordisk.com/prod-internal-4.3.1/latest"
   }
  ]
 },
```

Delete

renvlock-{trial}.txt

```
{
 "R": {
  "Version": "4.3.1",
  "Repositories": [
   {
    "Name": "CRAN",
    "URL": "https://rspm.bifrost-dev.corp.aws.novonordisk.com/default-cran/2023-10-25"
   }




  ]
 },
```

Point to Private
Package Manager

renvlock-{trial}.txt

```
{
  "R": {
    "Version": "4.3.1",
    "Repositories": [
      {
        "Name": "CRAN",
        "URL": "https://packagemanager.posit.co/cran/2023-10-25"
      }




    ]
  },
```

Point to Posit Public Package Manager

renvlock-{trial}.txt

Send to authorities

pkglite-{trial}.txt

Re-packaged NN packages

renvlock-{trial}.txt

`renv.lock` with reference to MRAN

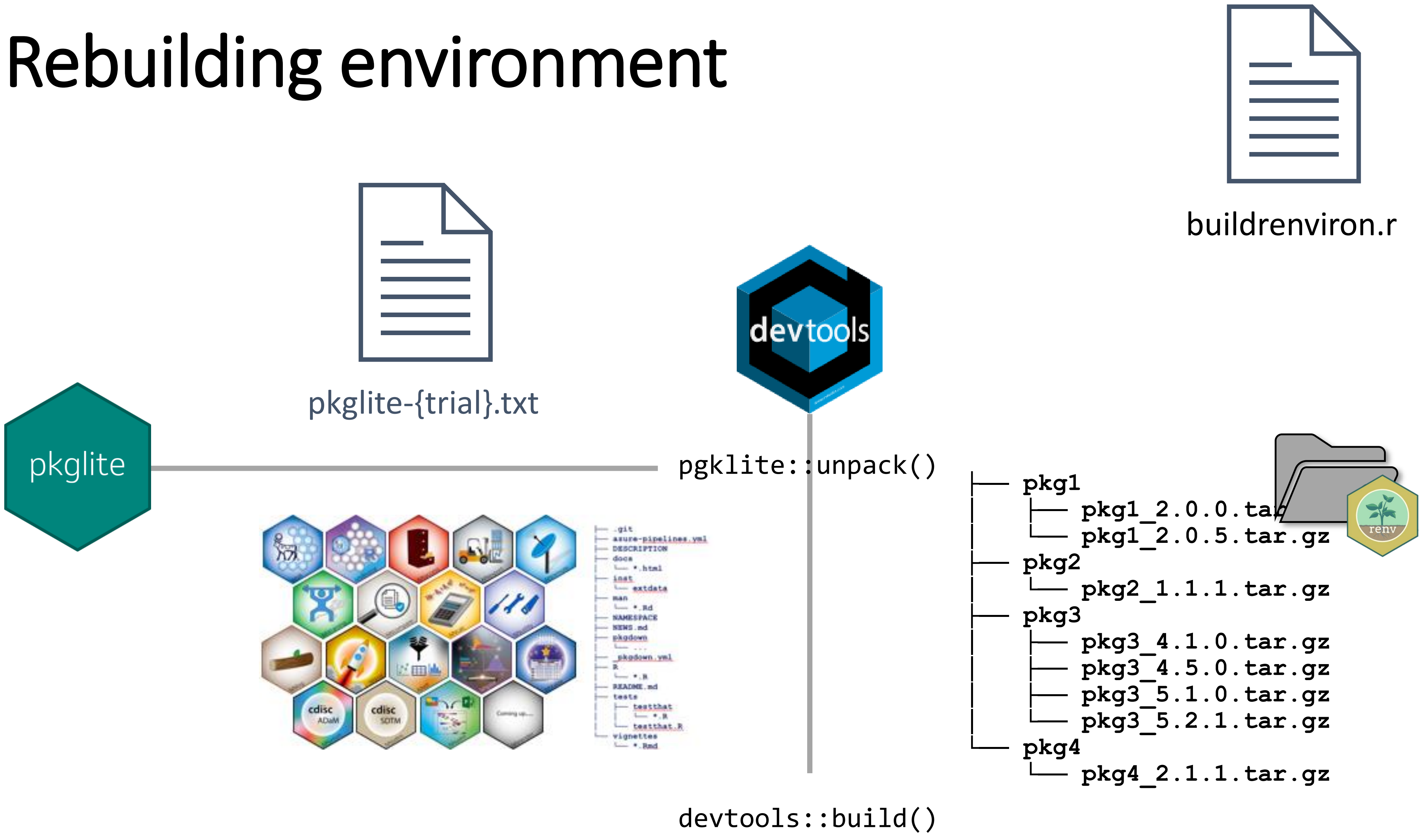
activate.txt

`renv` bootstrapping script

buildrenviron.r

Unpack, setup, rebuild and initialise environment

Rebuilding environment



pkglite-{trial}.txt

buildrenviron.r

pgklite::unpack()

devtools::build()

```
├── .git
├── azure-pipelines.yml
├── DESCRIPTION
├── docs
│   └── *.html
├── inst
│   └── extdata
├── man
│   └── *.Rd
├── NAMESPACE
├── NEWS.md
├── pkgdown
│   └── ...
├── _pkgdown.yml
├── R
│   └── *.R
├── README.md
├── tests
│   ├── testthat
│   │   └── *.R
│   └── testthat.R
└── vignettes
    └── *.Rmd
```

```
├── pkg1
│   ├── pkg1_2.0.0.tar
│   └── pkg1_2.0.5.tar.gz
├── pkg2
│   └── pkg2_1.1.1.tar.gz
├── pkg3
│   ├── pkg3_4.1.0.tar.gz
│   ├── pkg3_4.5.0.tar.gz
│   ├── pkg3_5.1.0.tar.gz
│   └── pkg3_5.2.1.tar.gz
└── pkg4
    └── pkg4_2.1.1.tar.gz
```

Rebuild environment

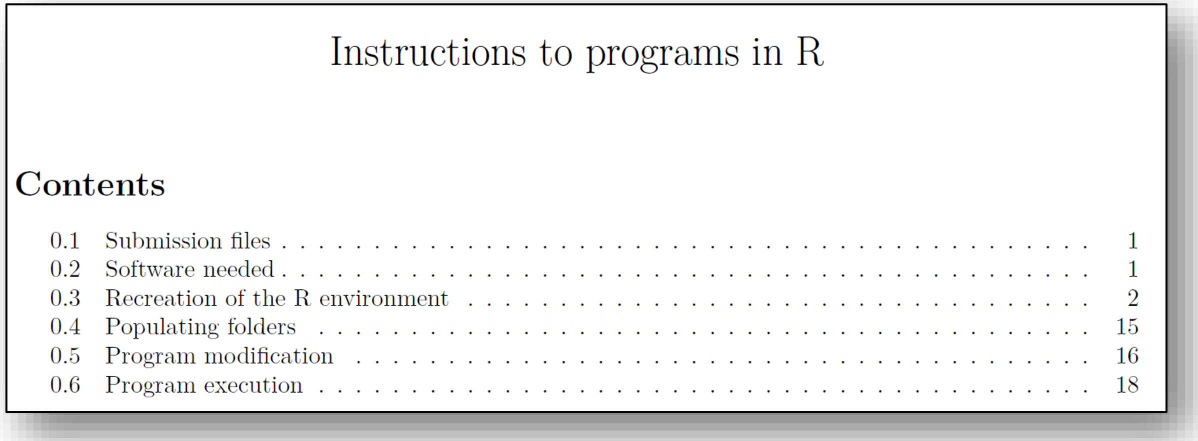
buildrenviron.r

renv::restore()

For each trial

```
├── project.Rproj
├── SDTM data
│   ├── ae.xpt
│   └── lb.xpt
├── ADaM data
│   ├── adsl.xpt
│   ├── adae.xpt
│   └── adlb.xpt
├── ADaM programs
│   ├── adsl.R
│   ├── adae.R
│   └── adlb.R
├── Output programs
│   ├── f_meanplot.R
│   └── t_adverse.R
├── Output
├── renv
│   ├── activate.R
│
├── .Renviron
├── .Rprofile
└── renv.lock
```

Rebuild environment



Instructions to programs in R

Contents

buildrenviron.r

renv::restore()

For each trial

```
├── project.Rproj
├── SDTM data
│   ├── ae.xpt
│   └── lb.xpt
├── ADaM data
│   ├── adsl.xpt
│   ├── adae.xpt
│   └── adlb.xpt
├── ADaM programs
│   ├── adsl.R
│   ├── adae.R
│   └── adlb.R
├── Output programs
│   ├── f_meanplot.R
│   └── t_adverse.R
├── Output
├── renv
│   ├── activate.R
│   └── library
├── .Renviron
├── .Rprofile
└── renv.lock
```

Thanks for joining