

Paquetes & Código de R

Julio 2019

Traducido por Leonardo Collado-Torres

@fellgernon

lcolladotor@gmail.com

lcolladotor.github.io

Desarrollado por Charlotte Wickham para rstudio::conf(2019)

@cvwickham

cwickham@gmail.com

cwick.co.nz

Adapted from *Tidy Tools* by Hadley Wickham



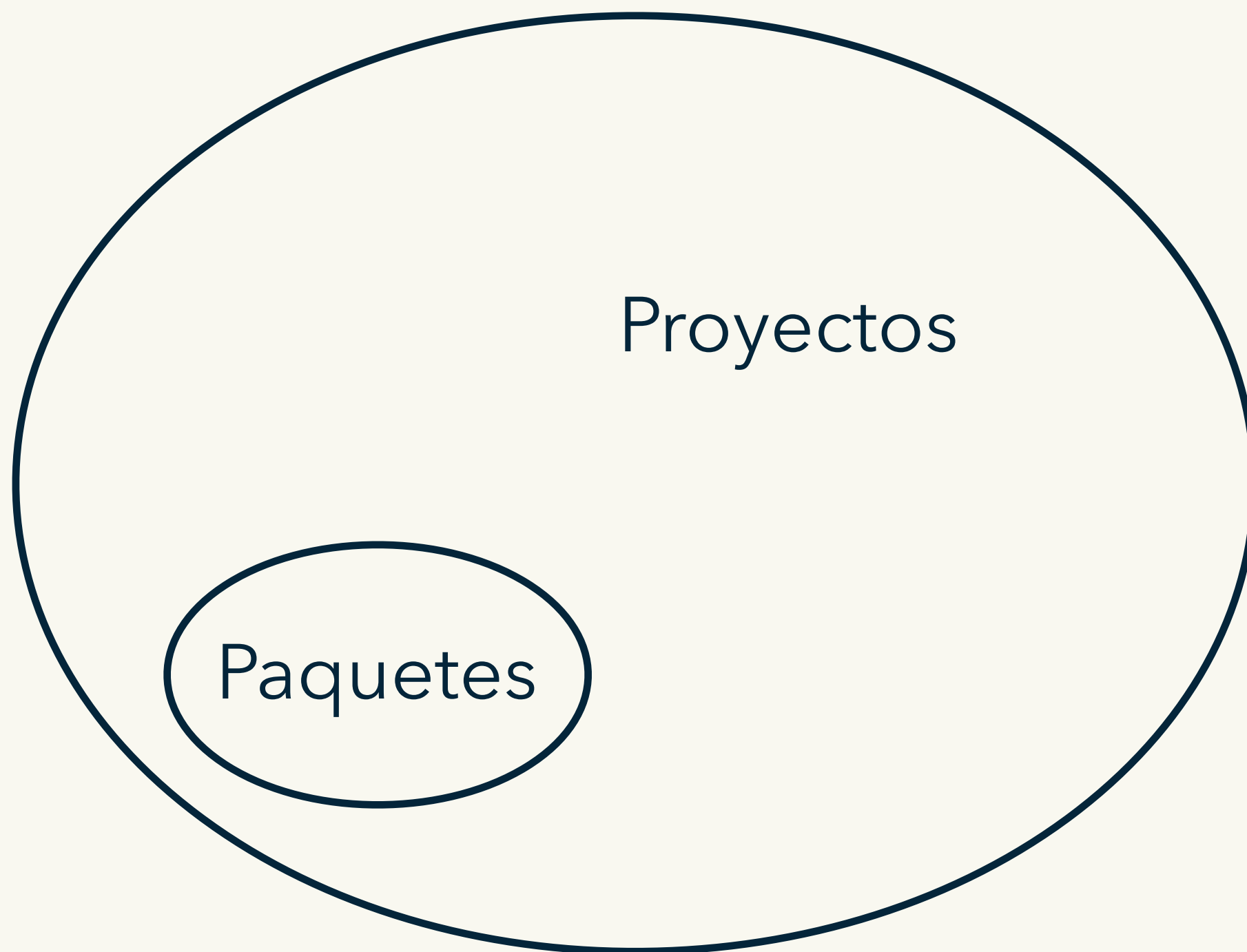
Motivación

Un paquete es una serie
de convenciones que
(con las herramientas
correctas)

hace tu vida más sencilla

“En serio, no tiene que ser sobre compartir tu código (¡aunque es un beneficio añadido!). Es sobre ahorrarte tiempo.”

– *Hilary Parker*

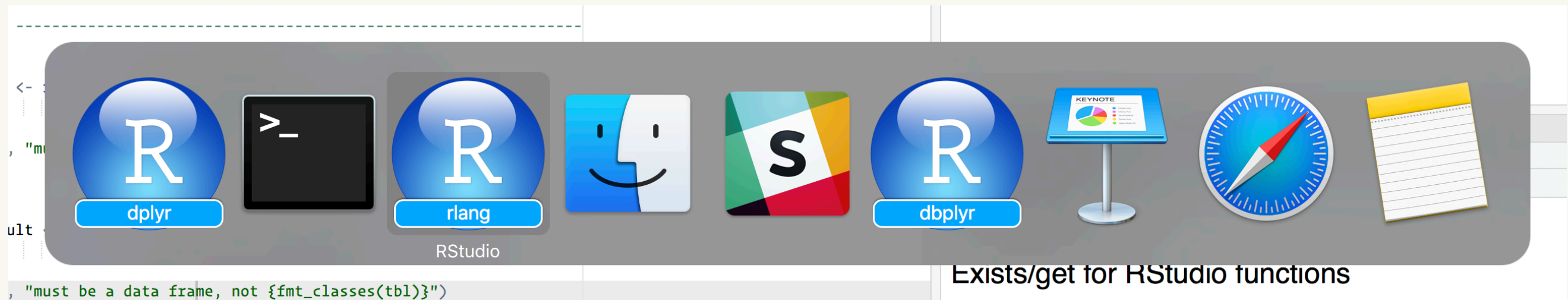


Proyectos de RStudio

¿Por qué usar proyectos de RStudio?

3 razones

Múltiples sesiones de Estudio



Trabaja en múltiples proyectos de forma simultánea e independiente

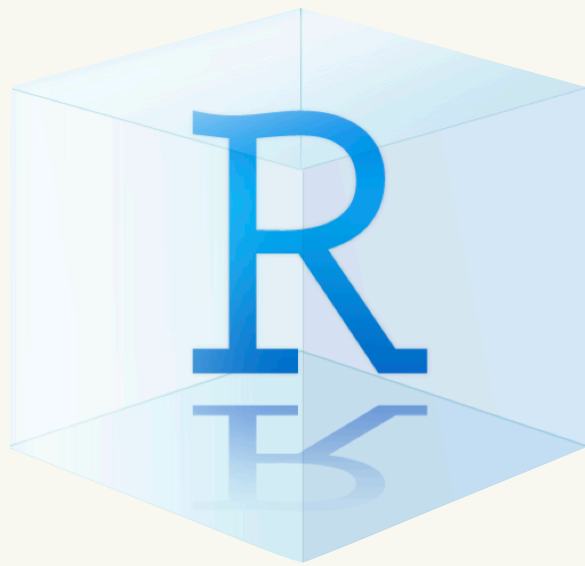
Organiza tus directorios de trabajo

Si la primera línea de tu script de R es
`setwd("C:\directorio\que\solo\yo\tengo")`
iré a tu oficina y LE PRENDERÉ FUEGO A
TU COMPUTADORA🔥.

– *Jenny Bryan*

<https://www.tidyverse.org/articles/2017/12/workflow-vs-script/>

Organiza tus directorios de trabajo

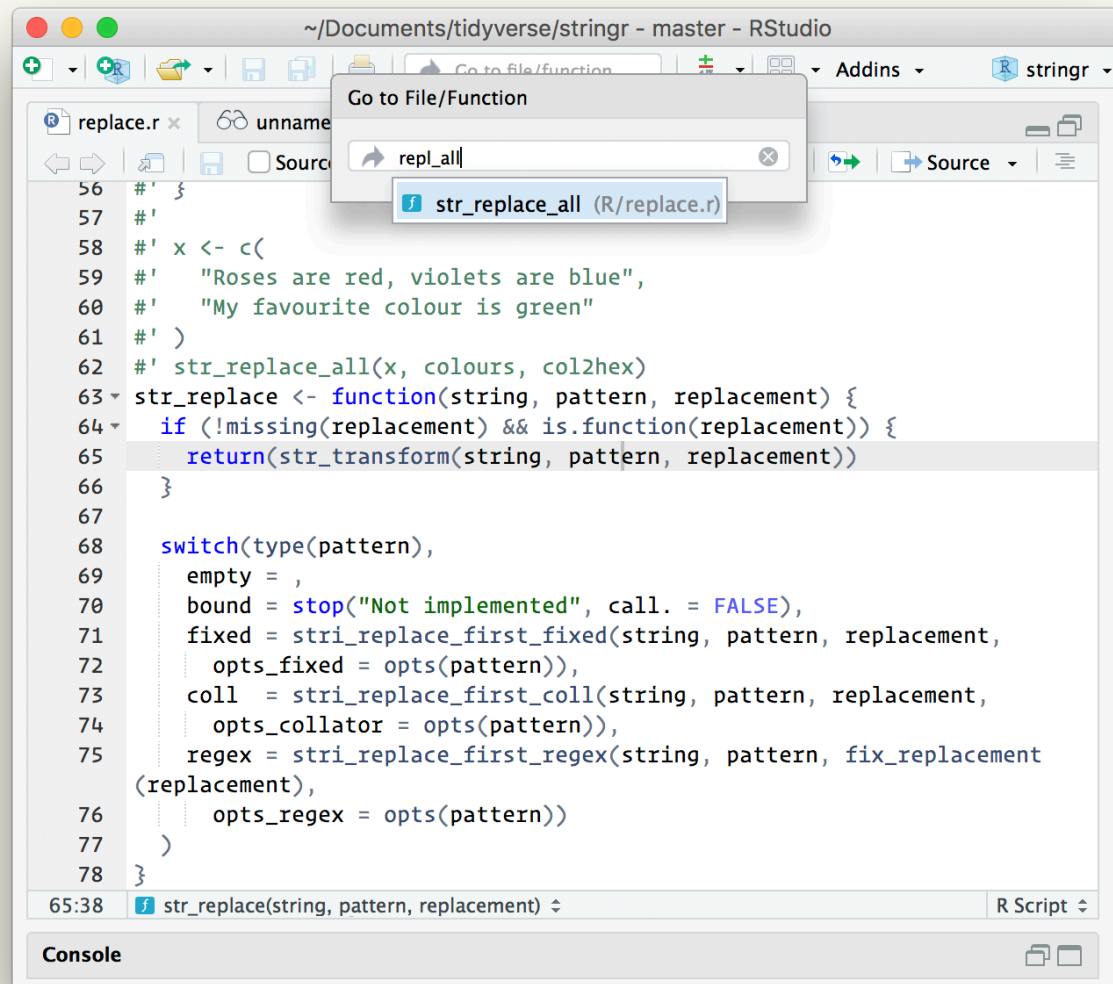


Dale doble click para abrir
este proyecto de RStudio

`mylittlepony.Rproj`

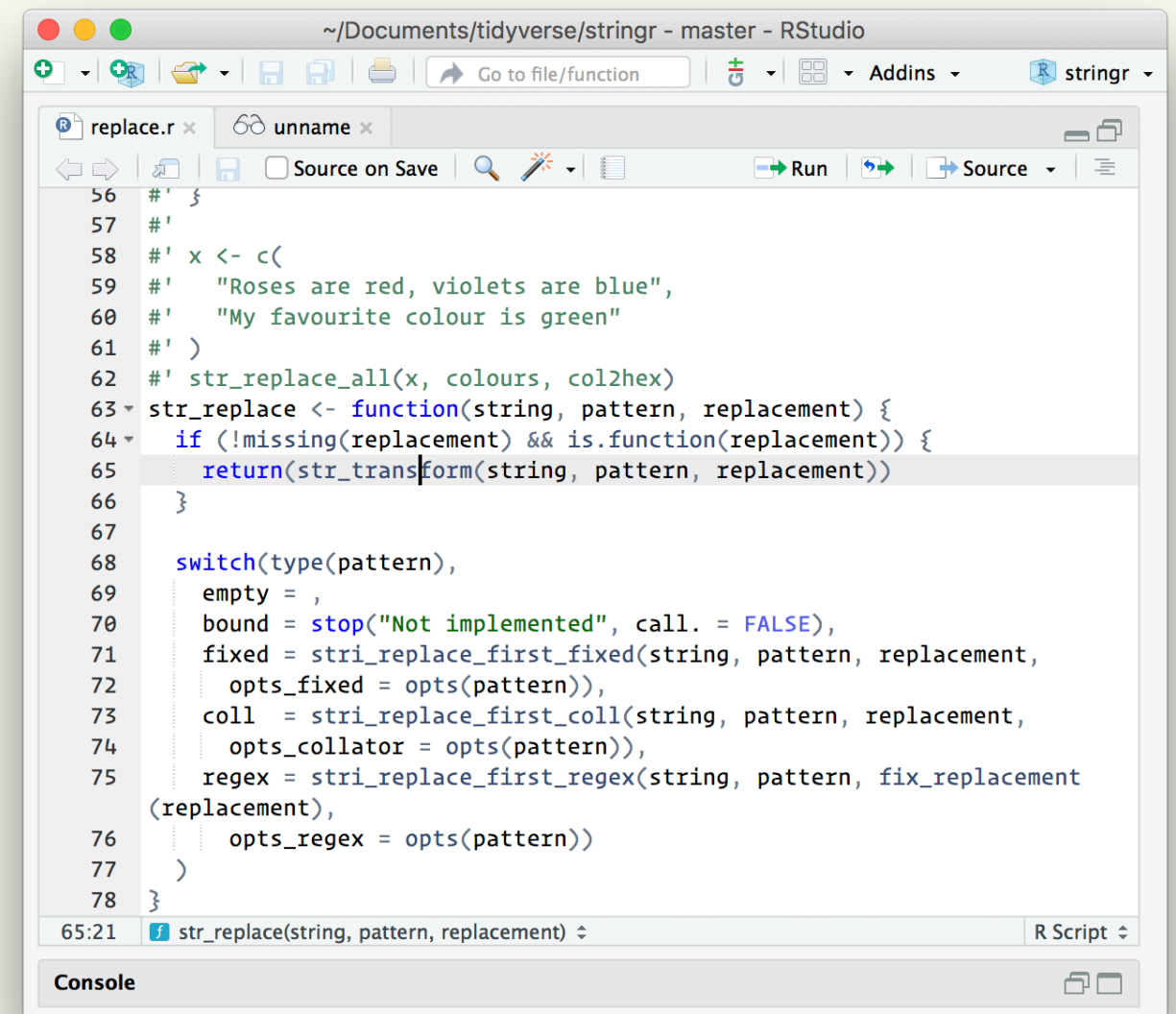
Navegación mejorada

Ctrl + . = encuentra
funciones/archivos



This screenshot shows the RStudio interface with the 'replace.r' file open. A 'Go to File/Function' dialog box is displayed, showing the search results for 'repl_all', with 'str_replace_all (R/replace.r)' selected. The code in the editor shows a function 'str_replace' and a comment for 'str_replace_all'.

```
56 #' }
57 #'
58 #' x <- c(
59 #'   "Roses are red, violets are blue",
60 #'   "My favourite colour is green"
61 #' )
62 #' str_replace_all(x, colours, col2hex)
63 str_replace <- function(string, pattern, replacement) {
64   if (!missing(replacement) && is.function(replacement)) {
65     return(str_transform(string, pattern, replacement))
66   }
67
68   switch(type(pattern),
69     empty = ,
70     bound = stop("Not implemented", call. = FALSE),
71     fixed = stri_replace_first_fixed(string, pattern, replacement,
72       opts_fixed = opts(pattern)),
73     coll = stri_replace_first_coll(string, pattern, replacement,
74       opts_collator = opts(pattern)),
75     regex = stri_replace_first_regex(string, pattern, fix_replacement
76       (replacement),
77       opts_regex = opts(pattern))
78   )
79 }
```



This screenshot shows the RStudio interface with the 'replace.r' file open. The cursor is at line 65, and the code is highlighted. The code in the editor shows a function 'str_replace' and a comment for 'str_replace_all'.

```
56 #' }
57 #'
58 #' x <- c(
59 #'   "Roses are red, violets are blue",
60 #'   "My favourite colour is green"
61 #' )
62 #' str_replace_all(x, colours, col2hex)
63 str_replace <- function(string, pattern, replacement) {
64   if (!missing(replacement) && is.function(replacement)) {
65     return(str_transform(string, pattern, replacement))
66   }
67
68   switch(type(pattern),
69     empty = ,
70     bound = stop("Not implemented", call. = FALSE),
71     fixed = stri_replace_first_fixed(string, pattern, replacement,
72       opts_fixed = opts(pattern)),
73     coll = stri_replace_first_coll(string, pattern, replacement,
74       opts_collator = opts(pattern)),
75     regex = stri_replace_first_regex(string, pattern, fix_replacement
76       (replacement),
77       opts_regex = opts(pattern))
78   )
79 }
```

F2 = brinca a la definición

Mi primer
paquete



`package.skeleton()`

¡Nunca usen esta función!

Tu turno

```
# Verifica que puedes crear un paquete usando:  
usethis::create_package("~/Desktop/mipaquete")
```

No uses esta parte si estás

usando rstudio.cloud

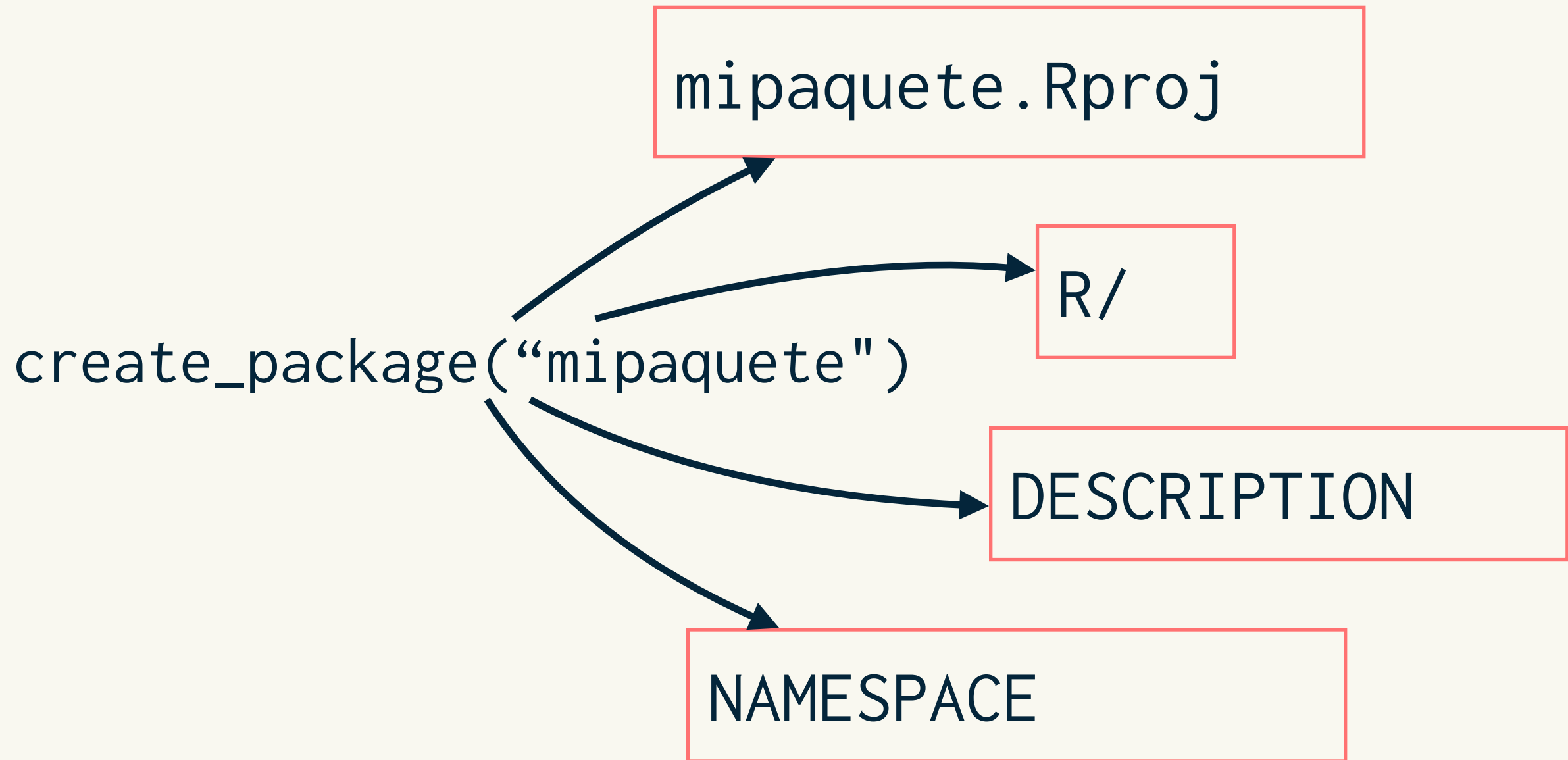
```
# ¿Que archivos y directorios fueron creados?
```

```
# También puedes crear un nuevo proyecto desde  
Estudio
```

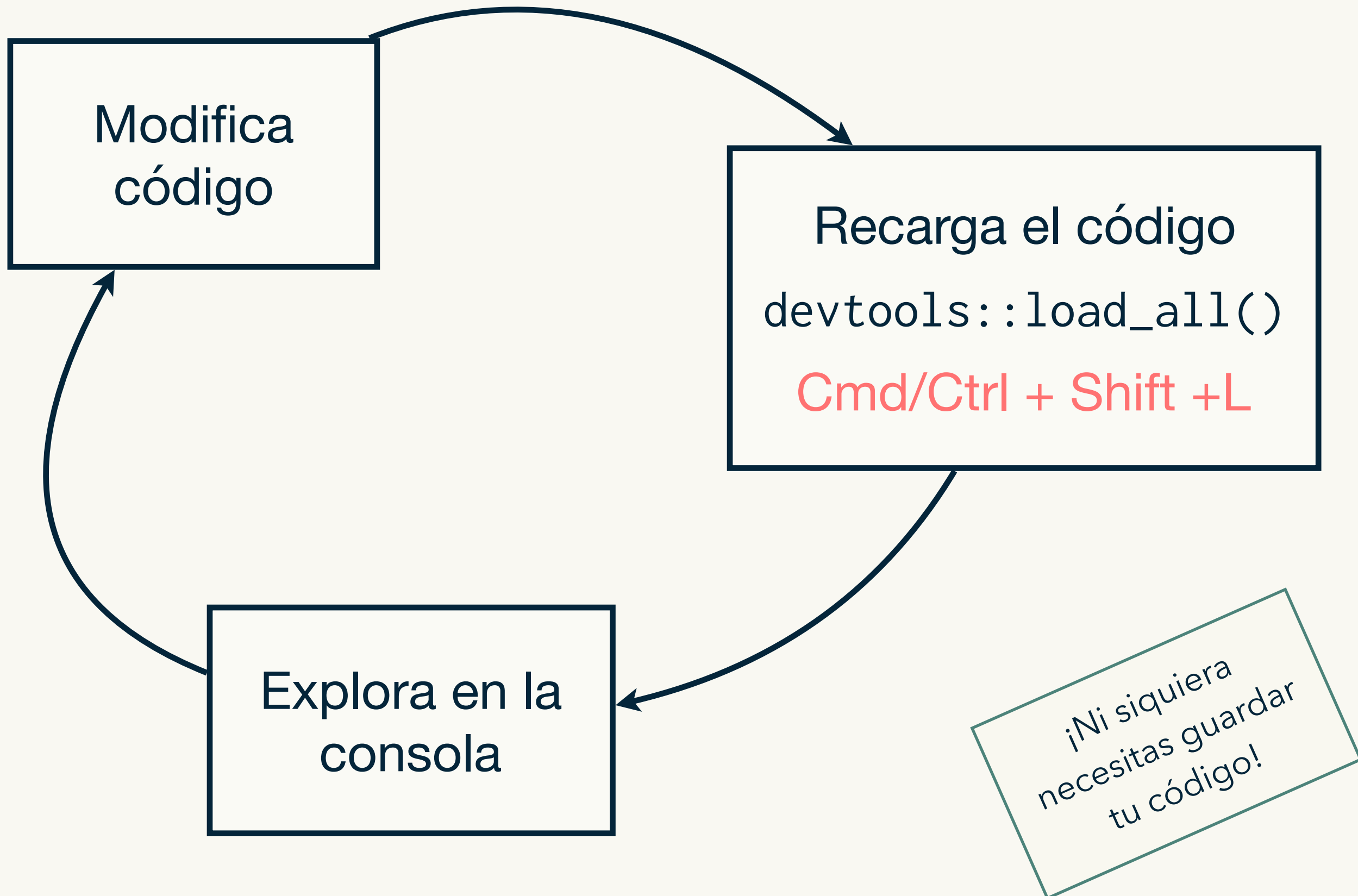
```
# pero tiene diferencias pequeñas que
```

```
# causarán fricción hoy (pero no en general)
```

¿Qué pasa cuando corremos `create_package()`?



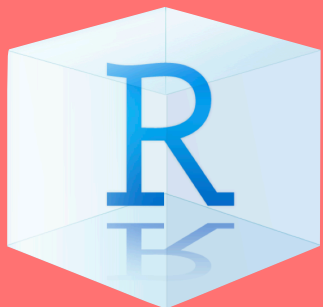
¿Para qué molestarnos? Mejor flujo de trabajo



Cambia al proyecto:

[mylittlepony]

tidytools/mylittlepony/



mylittlepony.Rproj

Doble click
para abrir

Tu turno

Navega a `rpony()` usando solo el teclado.

Carga todas las funciones (usando solamente el teclado), luego corre `rpony(10)`.

¡Uhoh! Se nos olvidó incluir a Fluttershy en la lista de ponies. Agrégala, recarga el código, y verifica que tu cambio haya funcionado.

¿Y si necesitas crear un archivo?

¡Hay una función en usethis para eso que nos ayuda!

```
usethis::use_r("nombre-del-archivo")
```

Organiza archivos de tal manera que
código relacionado vive junto. Si le puedes
dar a un archivo un nombre conciso e
informativo, probablemente está bien

Tu turno

Usa `usethis::use_r("show")` para crear un nuevo archivo

Agrega una función que imprime todos los ponies en orden alfabético.

Practica usando los atajos en el teclado.

Configuración de R/RStudio


Configuración del flujo de trabajo: tu .Rprofile

```
# Configura código que corre todas las veces  
# que abres R: usethis::edit_r_profile()
```

```
if (interactive()) {  
  suppressMessages(require(devtools))  
  suppressMessages(require(usethis))  
  suppressMessages(require(testthat))  
}
```

Nunca incluyas paquetes de análisis aquí

```
if (interactive()) {  
  suppressMessages(require(ggplot2))  
  suppressMessages(require(dplyr))  
}
```



Mientras estas ahí, también agrega

```
options(  
  warnPartialMatchArgs = TRUE,  
  warnPartialMatchDollar = TRUE,  
  warnPartialMatchAttr = TRUE  
)
```




Combina con **Ctrl/cmd + shift + F10**

Tu turno

Sigue las instrucciones en las diapositivas anteriores para asegurarte que estás configurado de forma óptima.

Lee *<https://www.tidyverse.org/articles/2017/12/workflow-vs-script/>* para mayor justificación/
motivación

Adapted from *Tidy Tools* by Hadley Wickham

This work is licensed as
Creative Commons
Attribution-ShareAlike 4.0
International

To view a copy of this license, visit
[https://creativecommons.org/
licenses/by-sa/4.0/](https://creativecommons.org/licenses/by-sa/4.0/)