boxr
a package to connect to CyBox

Ian Lyttle with Guillermo Basulto

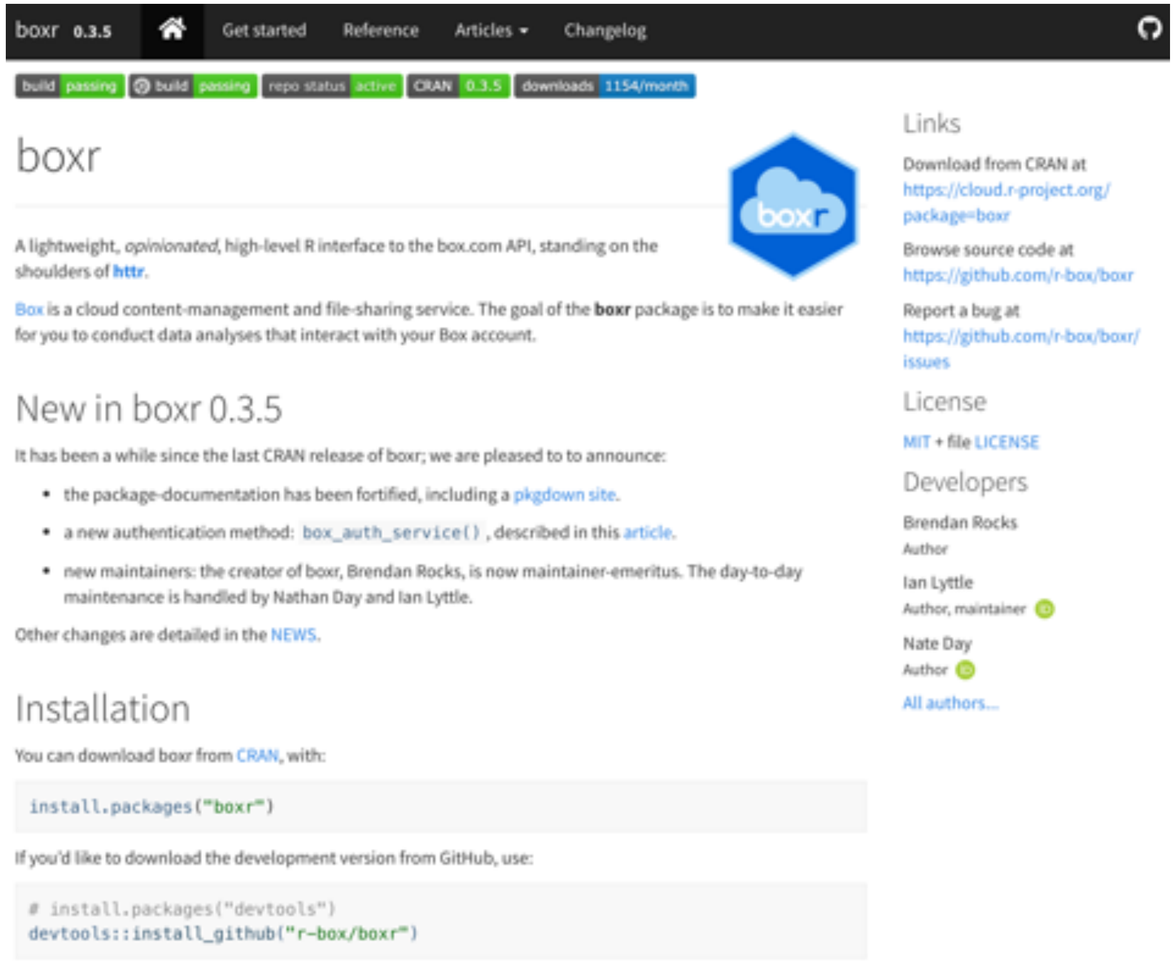
If you want to follow along:

```
> install.packages(c("devtools", "usethis"))
> devtools::install_github("r-box/boxr")
> usethis::use_course("ijlyttle/boxr-demo")
  # asks to open the boxr-demo project on your desktop
```

boxr

- CyBox is Box
- Created by Brendan Rocks
- Now maintained by Ian Lyttle and Nate Day
- Documentation:

  https://r-box.github.io/boxr

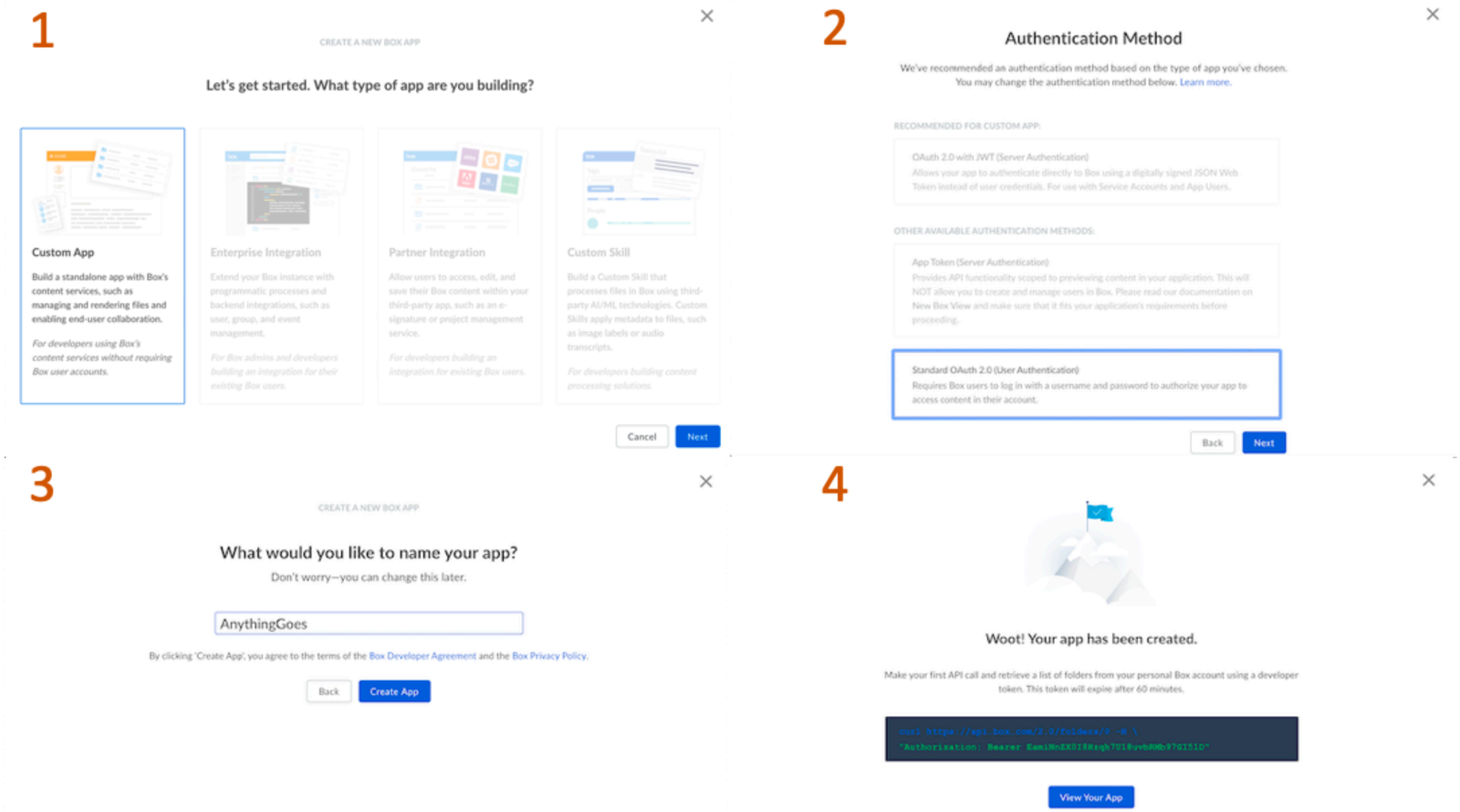


Authentication 1/3

First step is to create a Box App, which serves as the gateway between your Box account and the outside world:

https://app.box.com/developers/console

1. Select **Custom App**
2. Select **Standard OAuth 2.0**
3. Name your app
4. Go to **View your App**

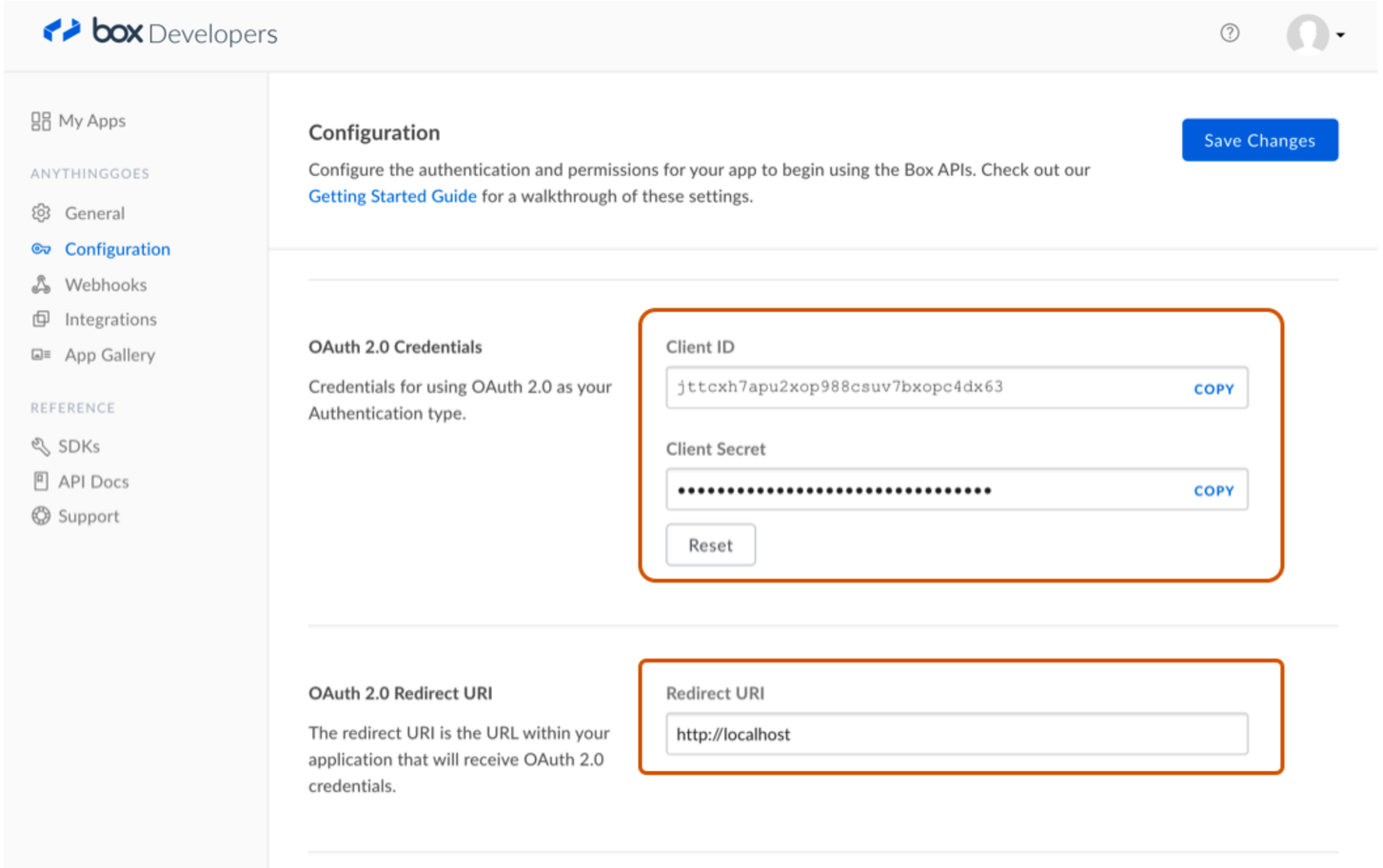


Authentication 2/3

Next step is to set OAuth2 parameters, go to **Configuration** panel:

1. Scroll down to **OAuth 2.0 Redirect URI**
2. Set **Redirect URI** to `http://localhost`
3. **Save Changes**

Copy:
`client_id`, `client_secret`



Authentication 3/3

Put the `client_id` and `client_secret` into your `.Renviron` file:

```
> usethis::edit_r_environ()
```

```
BOX_CLIENT_ID="client_id goes here"
BOX_CLIENT_SECRET="client_secret goes here"
```

Save, restart your R Session

```
> library("boxr")
> box_auth() # use this from now on...
```

"OAuth dance" takes you to a web page where you accept

Good news from there…

Box files and folders

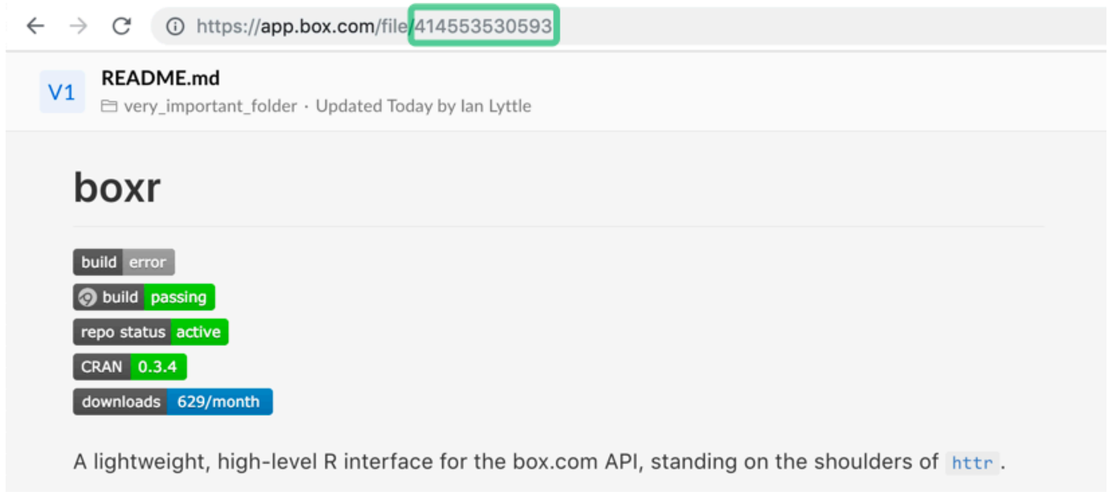
In boxr,

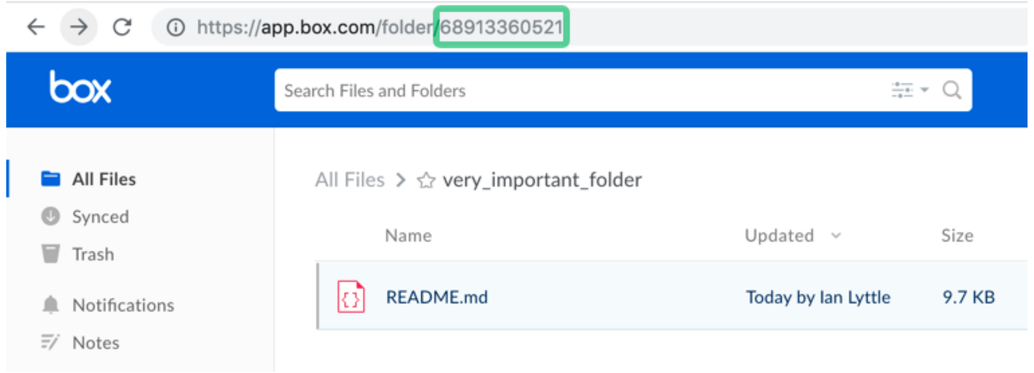
- files are identified using `file_id`
- folders are identified using `dir_id`

We use “dir” instead of “folder”:

  - How we inherited it
  - “dir” shorter than “folder”, more distinct from “file”
  - {fs} uses “dir”

You can get id values from the browser.





Demo: set up directories

```
> box_auth() # should be uneventful

> box_getwd() # should be 0

> box_dir_create("boxr-demo-data") # note dir_id

> box_ls() # see "boxr-demo-data" directory at Box

> box_setwd("dir_id of boxr-demo-data")

> setwd("data")
```

Demo: uploading to Box

```
> box_ul(file = "sample.csv")

> box_ls() # need to %>% to as.data.frame() to use

> box_dir_create("more-data")

> box_push("dir_id of more-data", local_dir = "more-data")

> box_ls()
```

Demo: make changes at Box

```
> box_browse(dir_id = "dir_id of more-data")

  # rename declaration.txt -> declaration-01.txt

> box_fetch("dir_id of more-data", local_dir = "more-data")

  # check out your local directory

> box_fetch("dir_id of more-data", local_dir = "more-data",
            delete = TRUE)

  # check out your local directory
```

boxr directory functions

`box_getwd()`, `box_setwd()` get, set Box working directory

`box_ls()` list files in Box working directory

`box_dir_create()` create Box directory

`box_fetch()`, `box_push()` recursively download, upload:

- `box_fetch()` acts more like “pull”
- take care with arguments `overwrite` and `delete`

boxr file functions

`box_ul()`, `box_dl()` upload, download Box file

Workflows

- GitHub is great for code, perhaps less great for data
- I use Box to separate code-storage from data-storage
- This lets me keep my code more-open and the data more-private.

Integrating into projects

```
very-important-project/
  00-get-data-from-box.Rmd
  01-wrangle-data.Rmd
  02-do-stuff.Rmd
  99-publish.Rmd
  data-private/
    00-from-box/
    01-wrangled/
    02-do-stuff/
  data-public/
```

Add `data-private/` to .gitignore

Retrieve reproducibly

Wrangle and do stuff reproducibly

Maybe publish data on GitHub

boxr keeps your credentials in your home directory

Share the directory with colleagues on Box - they can reproduce your work

Summary

- Biggest challenge is creating the Box app: `https://r-box.github.io/boxr/articles/boxr-apps.html`
- Once you get authentication worked out - it should *just work*. `box_auth()`
- Functions to interact with Box directories and files `file_id`, `dir_id`
- I like to store raw data on Box, as a complement to code on GitHub
  Keeps code open, raw data more private