

boxr

a package to connect to CyBox

Ian Lyttle with Guillermo Basulto

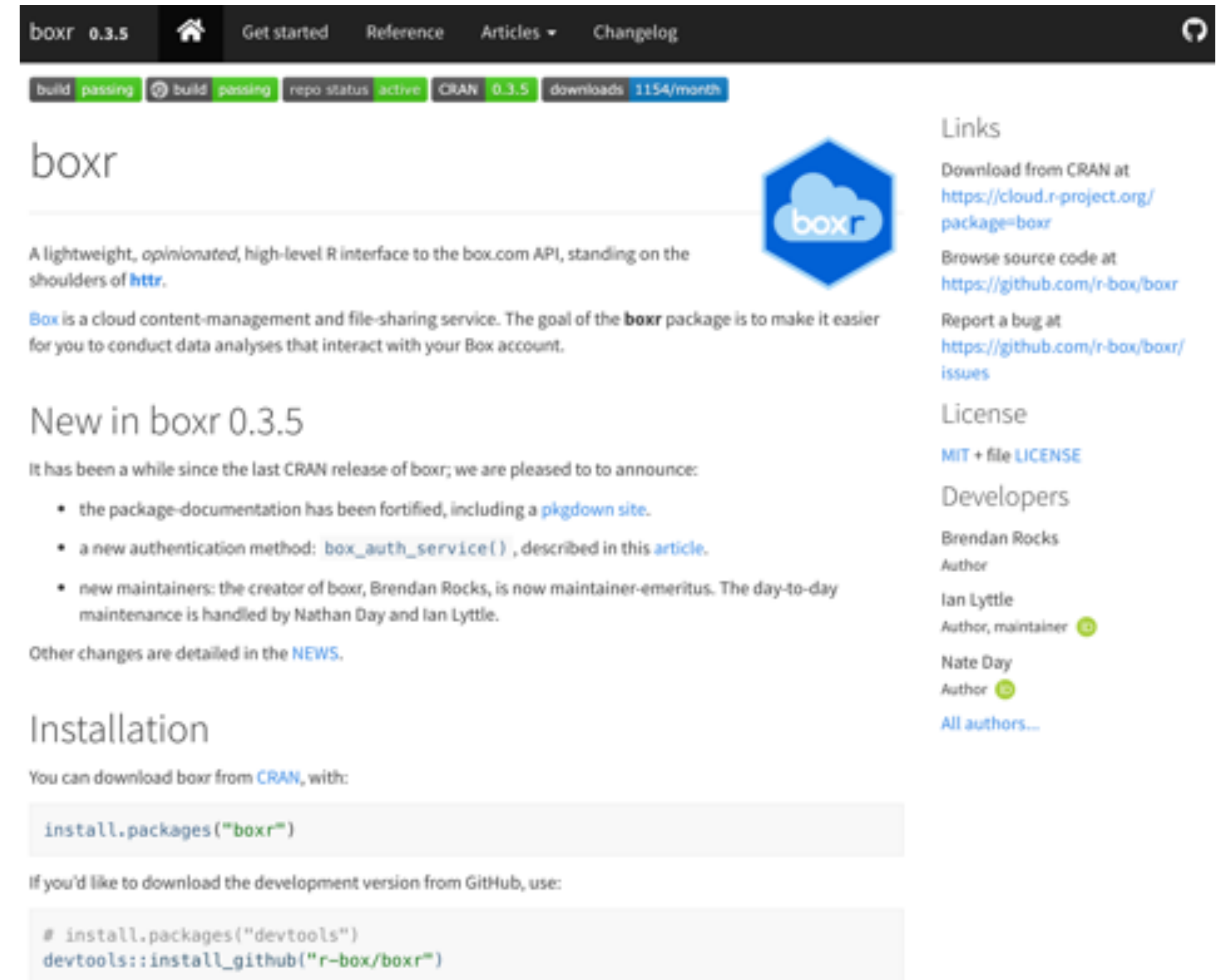
If you want to follow along:

```
> install.packages(c("boxr", "usethis"))  
> setwd("<a/good/directory>")  
> usethis::use_course("ijlyttle/boxr-demo", destdir = ".")
```

boxr

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

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



The screenshot shows the CRAN page for the **boxr** package. The header includes the package name and version (0.3.5), navigation links (Get started, Reference, Articles, Changelog), and status indicators (build passing, repo status active, CRAN 0.3.5, downloads 1154/month). The main content area features the boxr logo, a description of the package as a lightweight R interface to the box.com API, and a section titled "New in boxr 0.3.5" listing updates to documentation, authentication, and maintainers. An "Installation" section provides R code for installing the package from CRAN or GitHub. The right sidebar contains links to download from CRAN, browse source code, report bugs, and view the license and developers.

boxr 0.3.5

Get started Reference Articles Changelog

build passing build passing repo status active CRAN 0.3.5 downloads 1154/month

boxr

A lightweight, opinionated, high-level R interface to the box.com API, standing on the shoulders of [httr](#).

Box is a cloud content-management and file-sharing service. The goal of the **boxr** package is to make it easier for you to conduct data analyses that interact with your Box account.

New in boxr 0.3.5

It has been a while since the last CRAN release of boxr; we are pleased to to announce:

- the package-documentation has been fortified, including a [pkgdown site](#).
- a new authentication method: `box_auth_service()`, described in this [article](#).
- new maintainers: the creator of boxr, Brendan Rocks, is now maintainer-emeritus. The day-to-day maintenance is handled by Nathan Day and Ian Lyttle.

Other changes are detailed in the [NEWS](#).

Installation

You can download boxr from [CRAN](#), with:

```
install.packages("boxr")
```

If you'd like to download the development version from GitHub, use:

```
# install.packages("devtools")
devtools::install_github("r-box/boxr")
```

Links

Download from CRAN at <https://cloud.r-project.org/package=boxr>

Browse source code at <https://github.com/r-box/boxr>

Report a bug at <https://github.com/r-box/boxr/issues>

License

MIT + file [LICENSE](#)

Developers

Brendan Rocks
Author

Ian Lyttle
Author, maintainer

Nate Day
Author

[All authors...](#)

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**

The image displays four sequential screenshots of the Box Developer Console interface, numbered 1 through 4, illustrating the process of creating a new app.

Screenshot 1: Titled "CREATE A NEW BOX APP", it asks "Let's get started. What type of app are you building?". It offers four options: "Custom App" (highlighted with a blue border), "Enterprise Integration", "Partner Integration", and "Custom Skill". The "Custom App" option is described as building a standalone app with Box's content services.

Screenshot 2: Titled "Authentication Method", it shows recommended and other available authentication methods. "Standard OAuth 2.0 (User Authentication)" is selected and highlighted with a blue border. It requires Box users to log in with a username and password.

Screenshot 3: Titled "CREATE A NEW BOX APP", it asks "What would you like to name your app?". The input field contains "AnythingGoes". A "Create App" button is visible.

Screenshot 4: Titled "Woot! Your app has been created.", it shows a success message and a code snippet for the first API call. The code is:

```
curl https://api.box.com/2.0/folders/0 -H \n  "Authorization: Bearer Eam1Nr3X018Rqg7018uvb0Wb-970151D"
```

 A "View Your App" button is at the bottom.

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`

The screenshot shows the 'Configuration' panel in the Box Developers interface. The left sidebar contains a menu with 'My Apps', 'ANYTHINGGOES', 'General', 'Configuration' (highlighted), 'Webhooks', 'Integrations', 'App Gallery', 'REFERENCE', 'SDKs', 'API Docs', and 'Support'. The main content area is titled 'Configuration' and includes a 'Save Changes' button. Below this, there are two sections: 'OAuth 2.0 Credentials' and 'OAuth 2.0 Redirect URI'. The 'OAuth 2.0 Credentials' section contains fields for 'Client ID' (with a 'COPY' button) and 'Client Secret' (with a 'COPY' button and a 'Reset' button). The 'OAuth 2.0 Redirect URI' section contains a 'Redirect URI' field with the value 'http://localhost'.

box Developers

Configuration

Configure the authentication and permissions for your app to begin using the Box APIs. Check out our [Getting Started Guide](#) for a walkthrough of these settings.

Save Changes

OAuth 2.0 Credentials

Credentials for using OAuth 2.0 as your Authentication type.

Client ID

jttcxh7apu2xop988csuv7bxopc4dx63 COPY

Client Secret

..... COPY

Reset

OAuth 2.0 Redirect URI

The redirect URI is the URL within your application that will receive OAuth 2.0 credentials.

Redirect URI

http://localhost

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

```
> 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

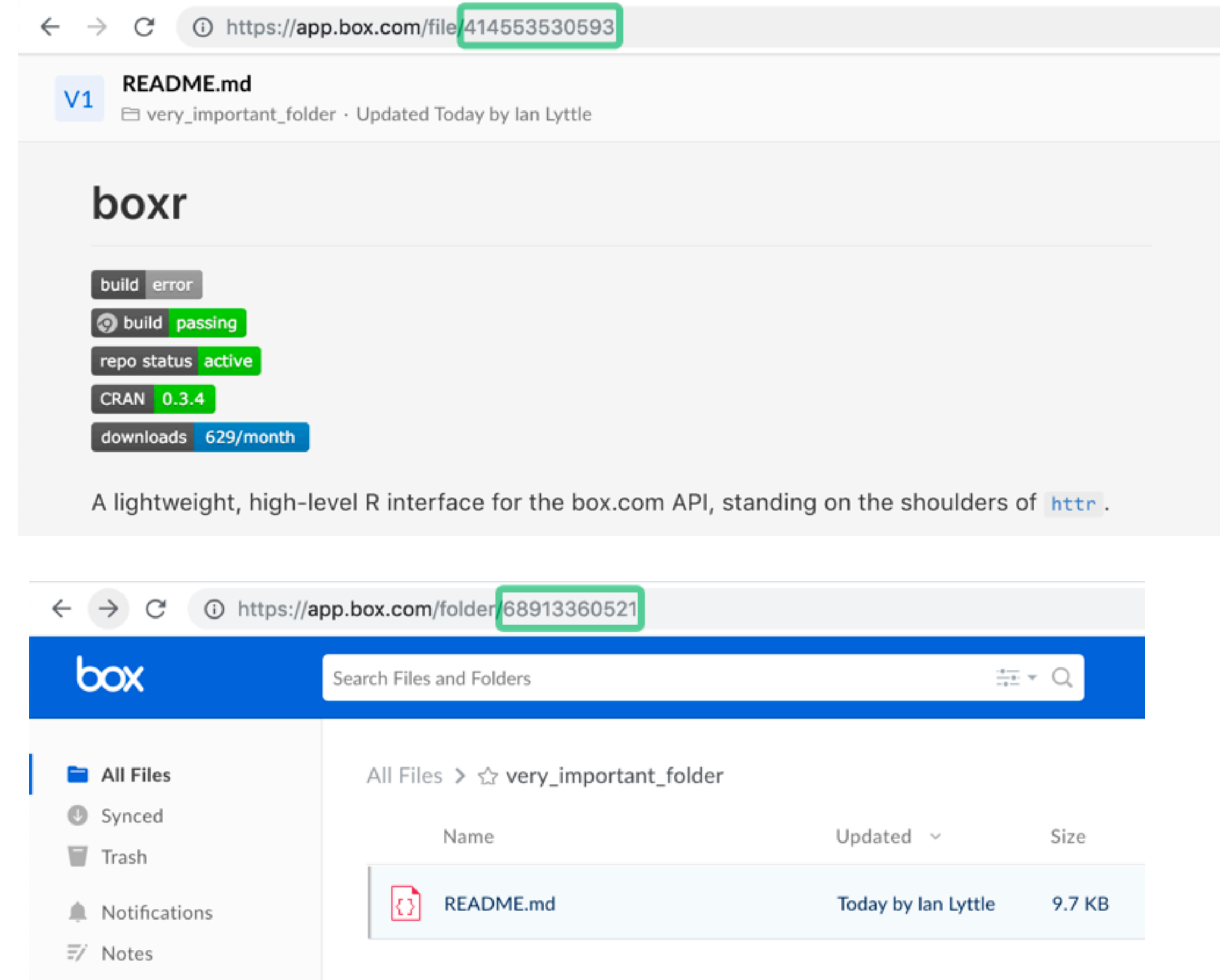
In boxr,

- 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 see get id values from the browser.



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.

Package Data

pkg/

R/

tests/

data/

data-raw/

data-from-box/

00-get-data-from-box.Rmd

01-wrangle-publish-data.Rmd

Add this to directory to .gitignore



```
graph LR; A[Add this to directory to .gitignore] --> D[data-from-box/]; B[Retrieve reproducibly] --> E[00-get-data-from-box.Rmd]; C[Wrangle and publish reproducibly] --> F[01-wrangle-publish-data.Rmd]
```

Retrieve reproducibly

Wrangle and publish reproducibly

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