An R client for iRODS

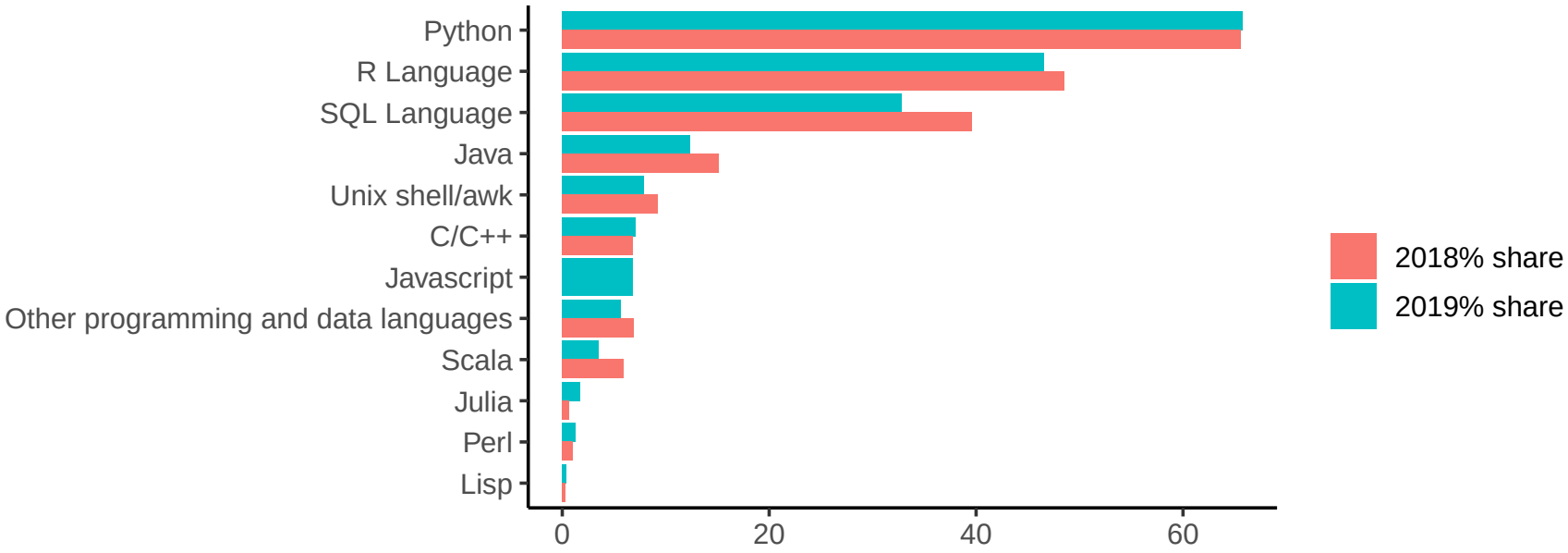
rirods

Martin Schobben Mariana Montes Christine Staiger
Terrell Russell

The R Ecosystem

Introduction to R

- Emphasis on statistics and visualization of data
- Used by researchers and industry
- Open sourced, active useRs community (26,616 packages)



Why R?

- Creating reproducible workflows
  - Scripted analysis
  - Literate programming ("Rmarkdown" and "Quarto")

Never again wonder what method did I use to center variable "foo" in my regression model ... ?

- But what about the data itself?
  - Centralized, relational, tabular databases

SQLite, MySQL, PostgreSQL, MonetDB with `DBI` package

Why iRODS?

- Freedom from strict formatting requirements
- Less data transformations mean higher productivity

```r
# height (cm)
x <- c(151, 174, 138, 186, 128, 136, 179, 163, 152, 131)
# weight (kg)
y <- c(63, 81, 56, 91, 47, 57, 76, 72, 62, 48)
# linear regression body mass index
BMI <- lm(y ~ x)
summary(BMI)
```

```
Call:
lm(formula = y ~ x)

Residuals:
    Min      1Q  Median      3Q     Max
-6.3002 -1.6629  0.0412  1.8944  3.9775
```

Why iRODS?

- Describing your data with metadata tags
- Making it findable for your peers

What was object `BMI` again?

```
ils(metadata = TRUE)
```

```
========
metadata
========
/tempZone/home/martin/BMI.rds :
 attribute            value units
  R object body mass index    lm


==========
iRODS Zone
==========
```

Designing an R package

CRAN Policies

Comprehensive R Archive Network (CRAN)

- The philosophy
  - Portablility: *Happy useRs across the board*
  - Stability: *Stringent requirements ensures a stable ecosystem*
- What constitutes a good package?
  - Tested and well-documented code
  - `R CMD check` 50+ tests

A Short History of R + iRODS

- Old R package build on the iRODS C++ API (archived)
- New R package build on the iRODS REST API

Feature\API	iRODS REST	iRODS C++
Portability		
Stability		

Global Design

- Mimic iCommands
- User facing
- Modular and adaptable (e.g. new REST API)

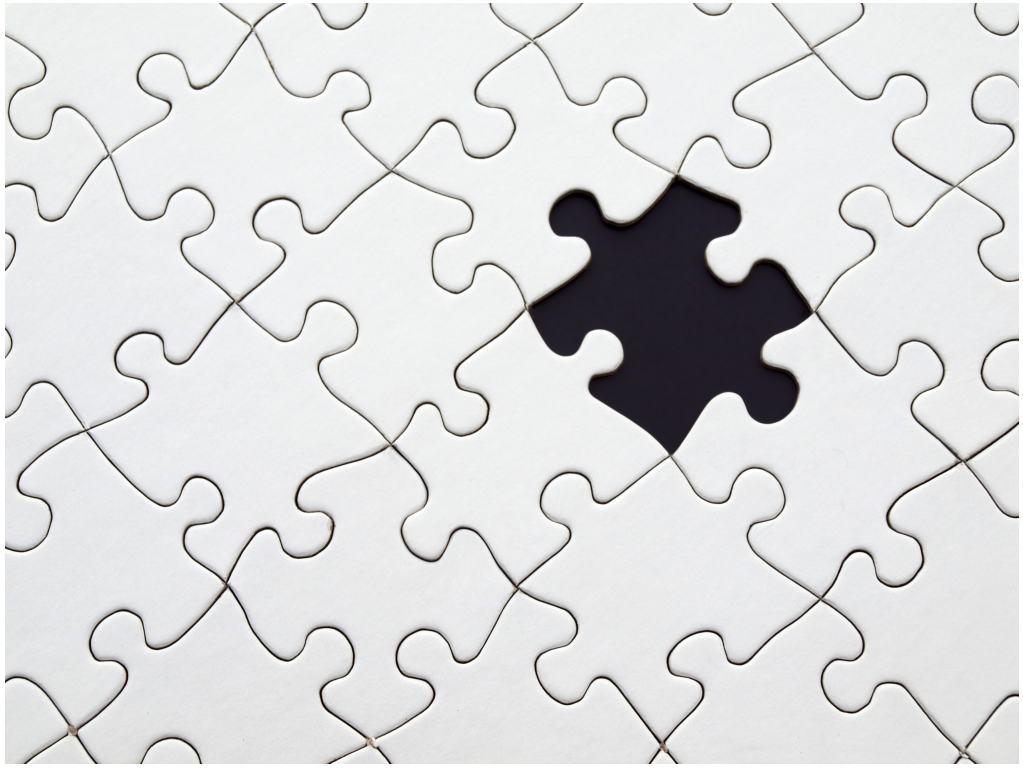


Figure 1: Photo from pexels.com

Interface

	R	iCommands
Authentication	`iauth`	`iinit`
Navigation	`icd`, `ils`, `ipwd`	`icd`, `ils`, `ipwd`
Objects	`iput`, `iget`, `imkdir`, `irm`, `isaveRDS`, `ireadRDS`	`iput`, `iget`, `imkdir`, `irm`
Metadata	`imeta`, `iquery`	`imeta`, `iquest`

Implementation

- Curl in R
  - R interface to libcurl *curl* (Ooms 2023a)
  - Wrapper *httr2* (Wickham 2023) for *curl* and *jsonlite* (Ooms 2023b)
- Development + Testing
  - iRODS demo-server
    - Terminal: `docker-compose up -d nginx-reverse-proxy`
    - R console: `use_irods_demo()`
  - Testing with mocking *httptest2* (Richardson 2022)
  - Automatic updates of snapshots with GitHub actions
  - `R CMD check` without internet (simulate CRAN checks)

Maintenance

- Source files on the iRODS GitHub organization page
- Website: https://irods.github.io/irods_client_library_rirods
- Maintainers
  - Martin Schobben, Vienna University of Technology, Austria
  - Mariana Montes, KU Leuven, Belgium

Future

- Submitted to CRAN
  `install.packages("rirods")`
- Publication of blog post on updates “iRODS4R”
- Upgrade in server side buffer size REST API to several Mb

Demonstration

Requirements:

- Remote iRODS server with iRODS C++ REST 0.9.3
- Demo server which requires `docker` and `docker-compose`
- `>= R 4.1`

Case study:

- Data set on iRODS commit history
- https://github.com/FAIReLABS/iRODS4R/blob/main/posts/welcome/data/irods_repos.csv

References

Ooms, Jeroen. 2023a. *Curl: A Modern and Flexible Web Client for r*. https://CRAN.R-project.org/package=curl.

———. 2023b. *Jsonlite: A Simple and Robust JSON Parser and Generator for r*. https://CRAN.R-project.org/package=jsonlite.

Richardson, Neal. 2022. *Httptest2: Test Helpers for Httr2*. https://CRAN.R-project.org/package=httptest2.

Wickham, Hadley. 2023. *Httr2: Perform HTTP Requests and Process the Responses*. https://CRAN.R-project.org/package=httr2.