R environment in GSK (WARP)

Tilo Blenk, Statistical Data Sciences, Biostatistics

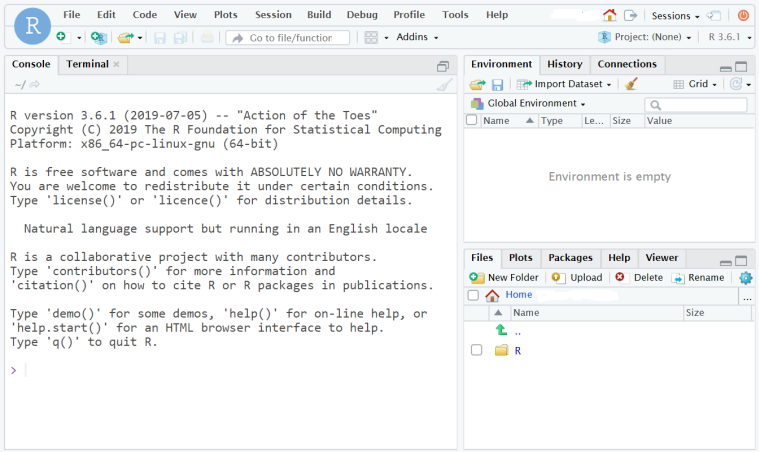
September 2020

Requirements for R environment

- Biostatistics use case:
  - statistical inference (plus causal inference and prediction modelling)
  - small to moderately sized structured, tabular data from clinical trials, eg SDTM/ADaM data sets
  - computational demanding statistical simulations and Bayesian computations
  - increasingly more analysis of molecular/genomics data
- functionalities to cover:
  - data analysis and software development
  - high performance computing (HPC)
  - reporting of analytic results, ie content sharing with R Markdown documents and Shiny web application
  - access to existing data sources like clinical trial data sets and molecular/genomics data
- centralised R computational environment
- capacity for about 1000 programmers/statisticians with 300 concurrent users

Language and tool support through R and RStudio

R and R packages



RStudio Server Pro

Shiny, R Markdown, plumber
RStudio Connect

git, GitHub Enterprise

Slurm, MPI
parallel, foreach, furrr

C++
Rcpp

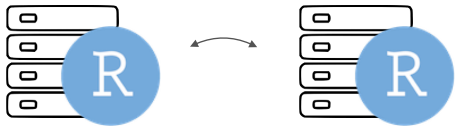
Python (Anaconda), reticulate

Stan
rstan, cmdstanr

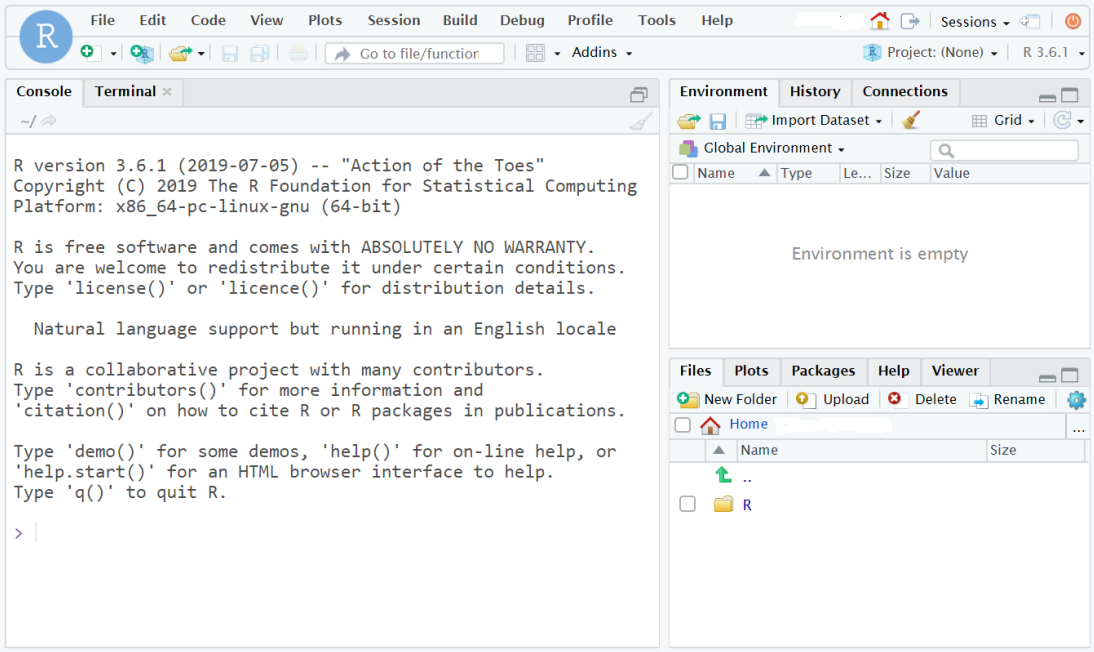
SQL
odbc
RSQLite

Java, rJava, RJDBC

Data analysis and software development



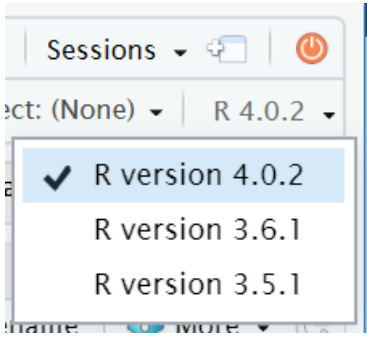
data analysis / software development
RStudio Server Pro



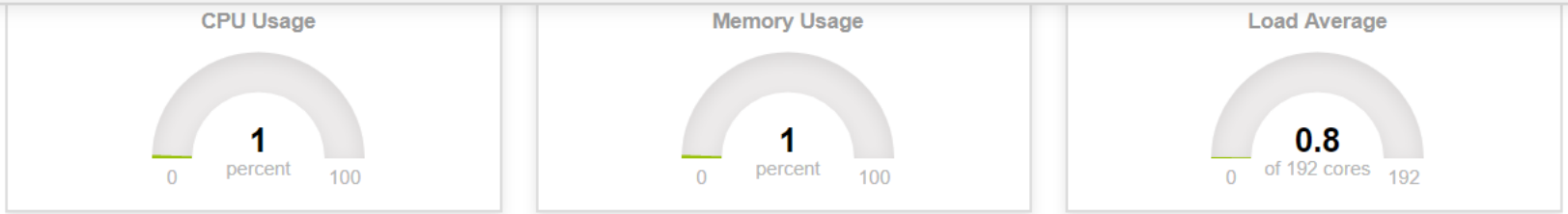
- 2 servers with RStudio Server Pro
  - to enable load balancing and high availability
  - to provide enough capacity for 300 concurrent users
- RStudio Server Pro as the integrated development environment (IDE)
  - support for multiple languages (edit and execute): R, Stan, C++, SQL, shell, Python
  - R package, Shiny application development support
  - graphical debugger and profiler for R
  - multiple R versions and sessions
  - integrated terminal
  - git integration with graphical user interface
- challenge: interactive data analysis together with parallel execution of HPC jobs

R installations and package management

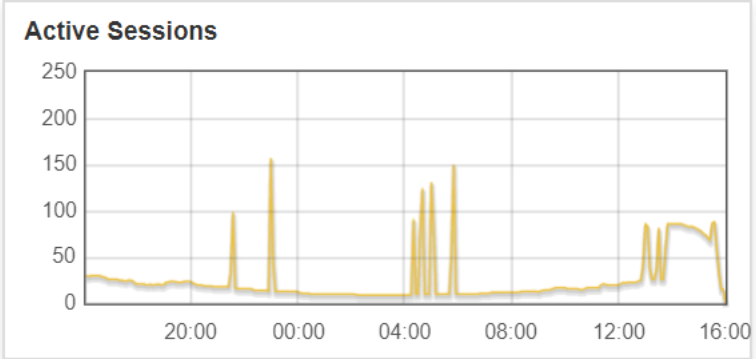
- multiple R installations available (eg R 4.0.2, R 3.6.1) consisting of
  - R base distribution
  - about 450 packages installed (as System Library) in their latest versions at time of installation
- R installations are not changed/updated, they are supposed to be reliable environments
- addition of 2 new R installations each year
  - one in early summer after new major version of R is released, eg 3.6, 4.0
  - another one at the end of a year, mainly for package updates
- at least the 6 latest R installations are planned to be available, ie reaching back 3 years
- users can choose which R installation to use for a session/project
- users can install additional packages or update packages in their User Library
- centralised implementation of R installations
- RStudio Package Manager with repos for CRAN, Bioconductor, and selected public/enterprise GitHub packages

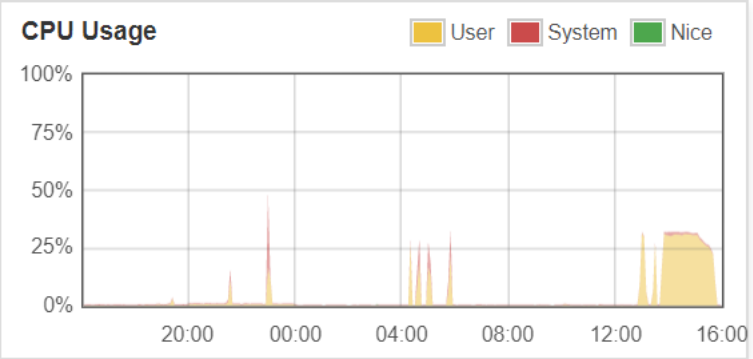


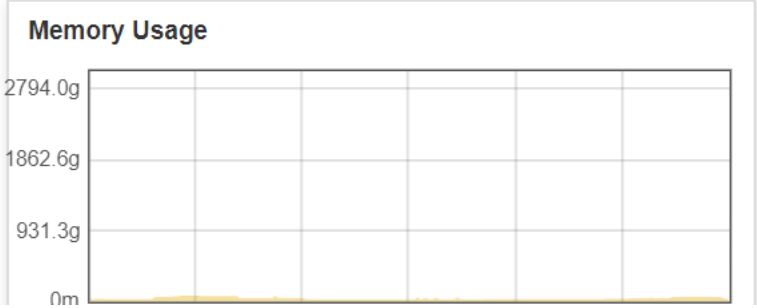
Core and memory usage

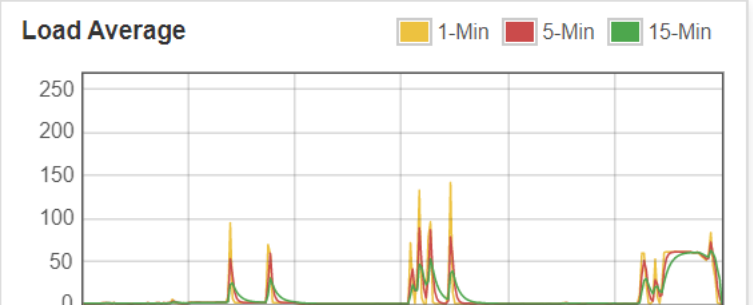


System History

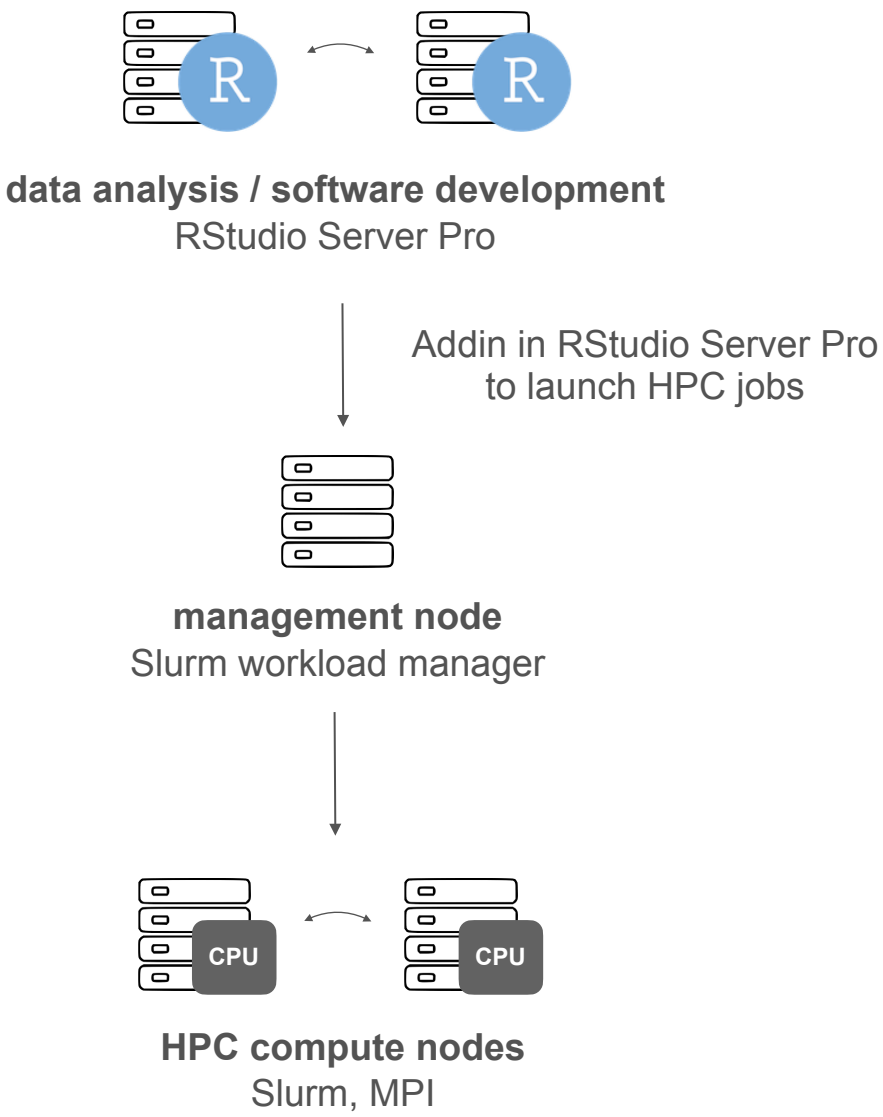






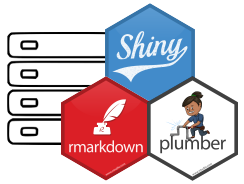


High performance computing (HPC)

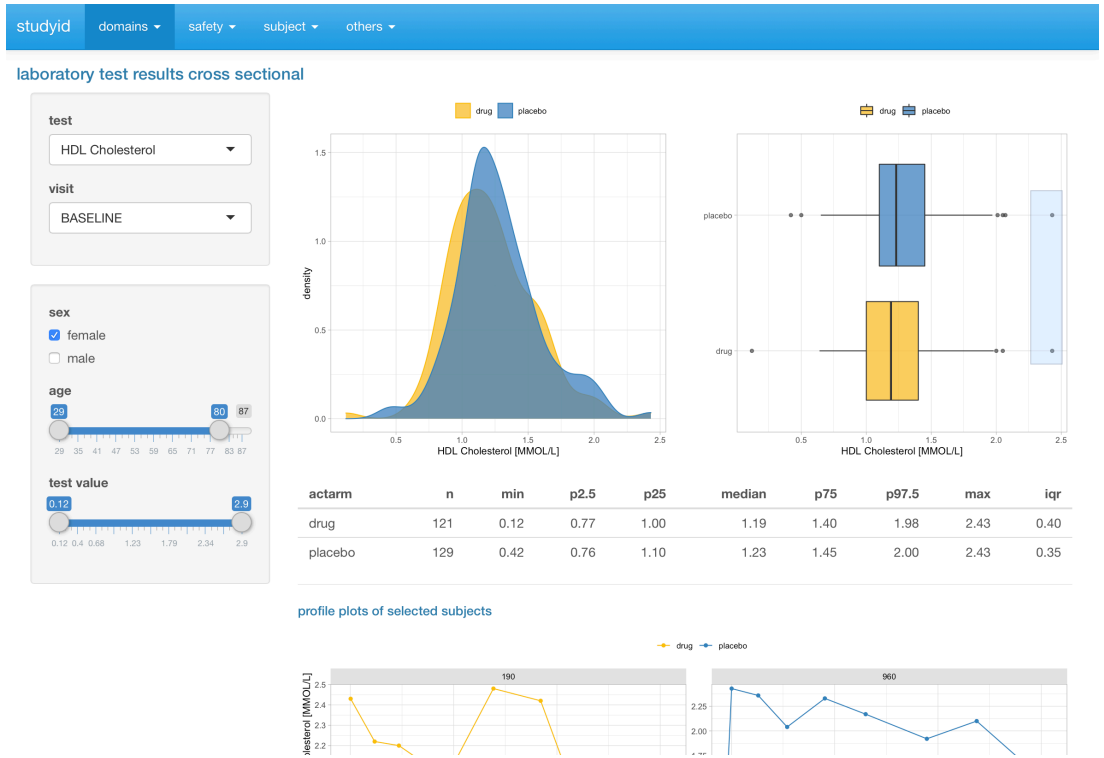


- 2 compute node
  (each with 192 logical cores and 3 TB memory)
- Slurm as job scheduler
- Addin in RStudio Server Pro to launch jobs on HPC compute nodes
- current usage:
  - parallelisation exclusively in R using parallel, foreach, furrr, rstan, cmdstanr
  - shared memory model
- guidance necessary:
  - how to parallelise and how to monitor execution
  - only short test runs with limited number of cores/ threads during development on RStudio Server Pro servers
  - execution of HPC jobs on compute nodes

Content sharing



content sharing
RStudio Connect



actarm	n	min	p2.5	p25	median	p75	p97.5	max	iqr
drug	121	0.12	0.77	1.00	1.19	1.40	1.98	2.43	0.40
placebo	129	0.42	0.76	1.10	1.23	1.45	2.00	2.43	0.35

- 1 server with RStudio Connect
- current usage:
  - documentation and training material
  - QDM app, Fundamentals of Stats app, ...
  - Shiny apps for reporting of clinical trial data
- currently liberal control model: users can publish what they want
- in future maybe additional RStudio Connect server with restricted publishing for regulated content
- Shiny applications as one main driver for interest in R and the computational environment
- guidance necessary for data access

Data backends

	use case	data example	interface
file shares	structured/unstructured data small size frequent updates read in total	source code files R scripts, R Markdown files	readr, jsonlite haven, readxl
object store	structured/unstructured small to big no frequent updates read in total	RNA-seq data FASTQ files	aws.s3
relational databases	structured tabular small to big frequent read/write often subset selection	clinical trial data SDTM/ADaM data	RSQLite, RPostgres odbc, RJDBC
Hadoop clusters	structured/unstructured big to huge mainly read (infrequent bulk loads) often subset selection often too big for in-memory processing	real world data electronic health records databases	sparklyr odbc, RJDBC

WARP architecture