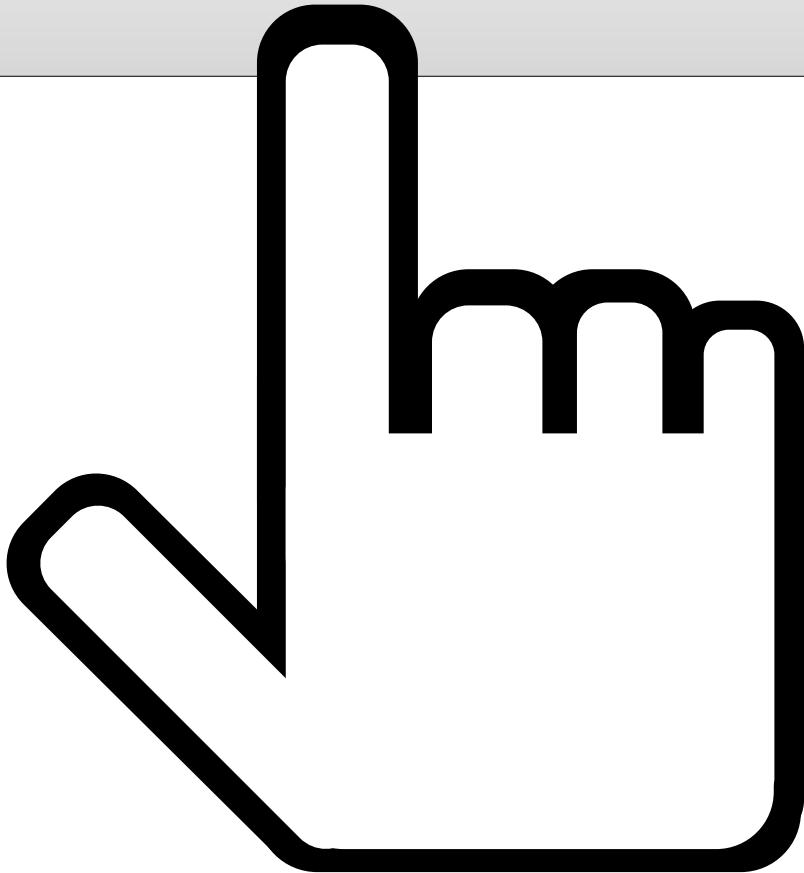


All Training materials are provided "as is" and without warranty and RStudio disclaims any and all express and implied warranties including without limitation the implied warranties of title, fitness for a particular purpose, merchantability and noninfringement.

The Training Materials are licensed under the Creative Commons Attribution-Noncommercial 3.0 United States License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc/3.0/us/> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

Welcome to **Introduction to Shiny**

 Run Document



Philip Bowsher
Customer Success
Sept 2017



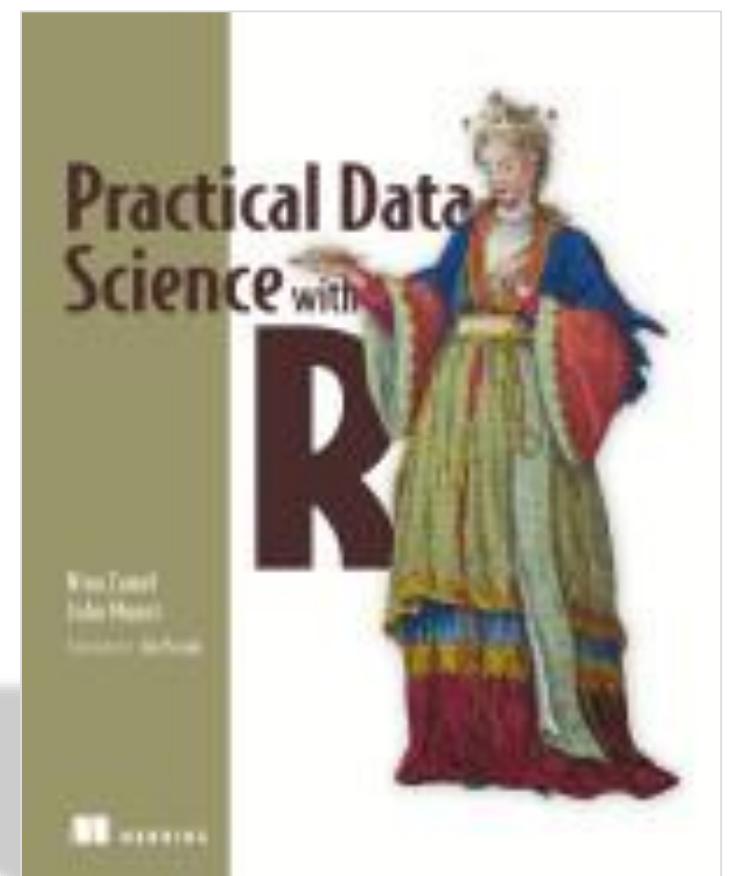
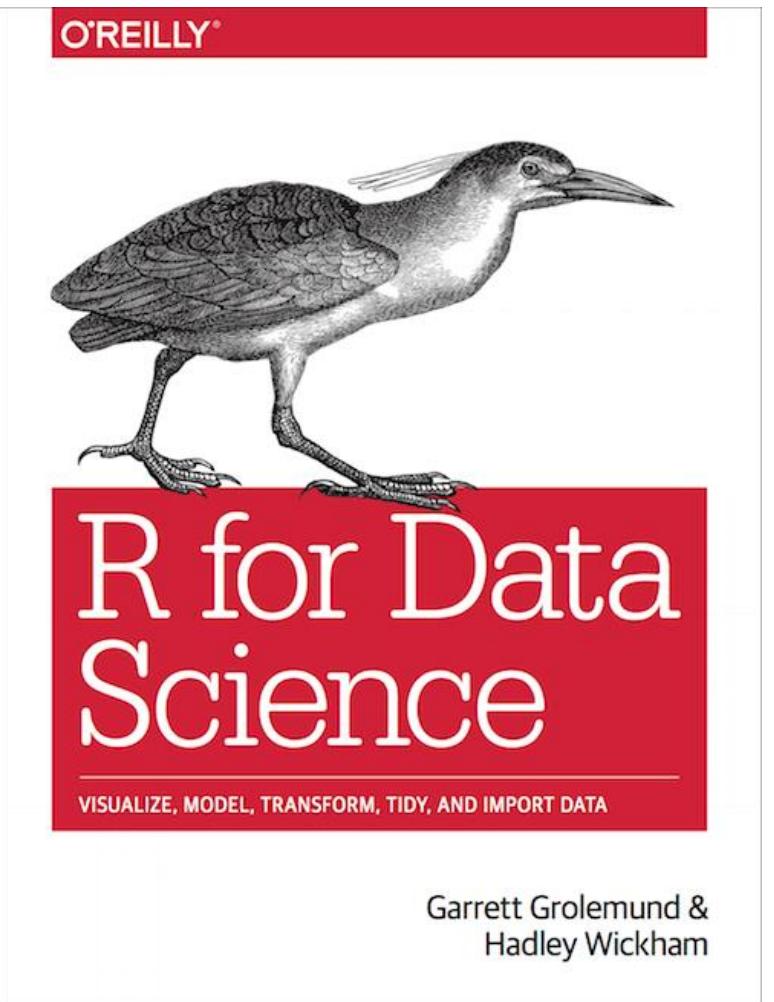
What is Data Science:

“Data science is an exciting discipline that allows you to turn raw data into understanding, insight, and knowledge. The goal of ‘R for Data Science’ is to introduce you to the most important in R tools that you need to do data science.”

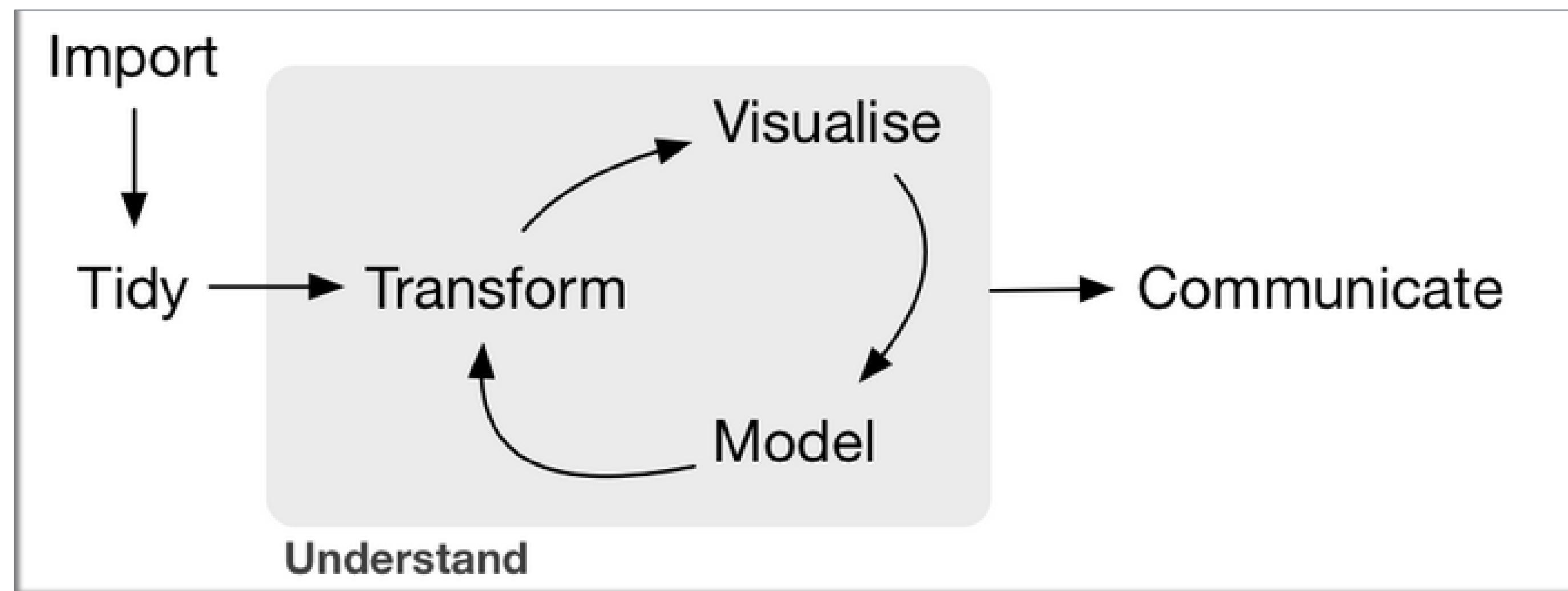
<http://r4ds.had.co.nz/intro.html>

“Data science is a term I use to represent the ownership and management of the entire modeling process: discovering the true business need, collecting data, managing data, building models and deploying models into production.”

<http://www.win-vector.com/blog/2013/04/data-science-machine-learning-and-statistics-what-is-in-a-name/>

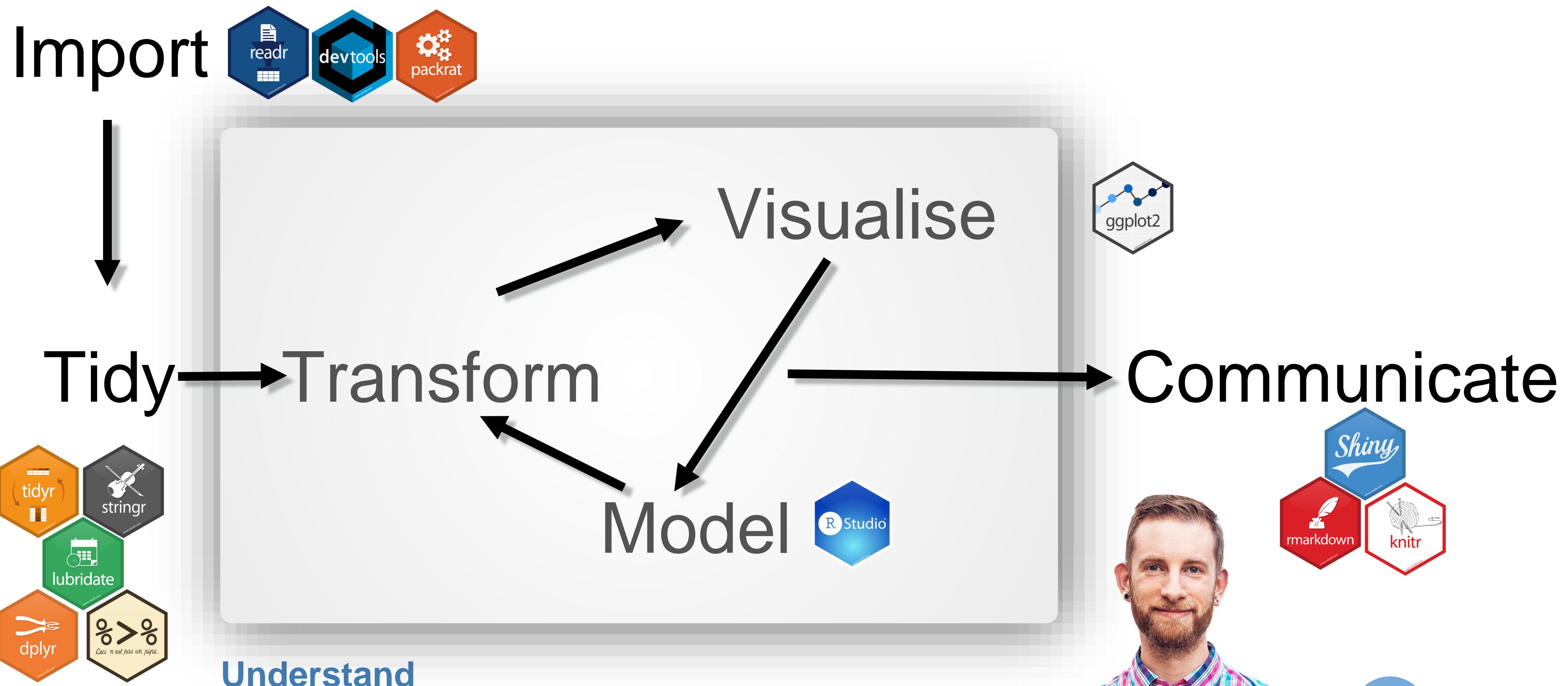


Model of the Tools Needed in a Typical Data Science Project :

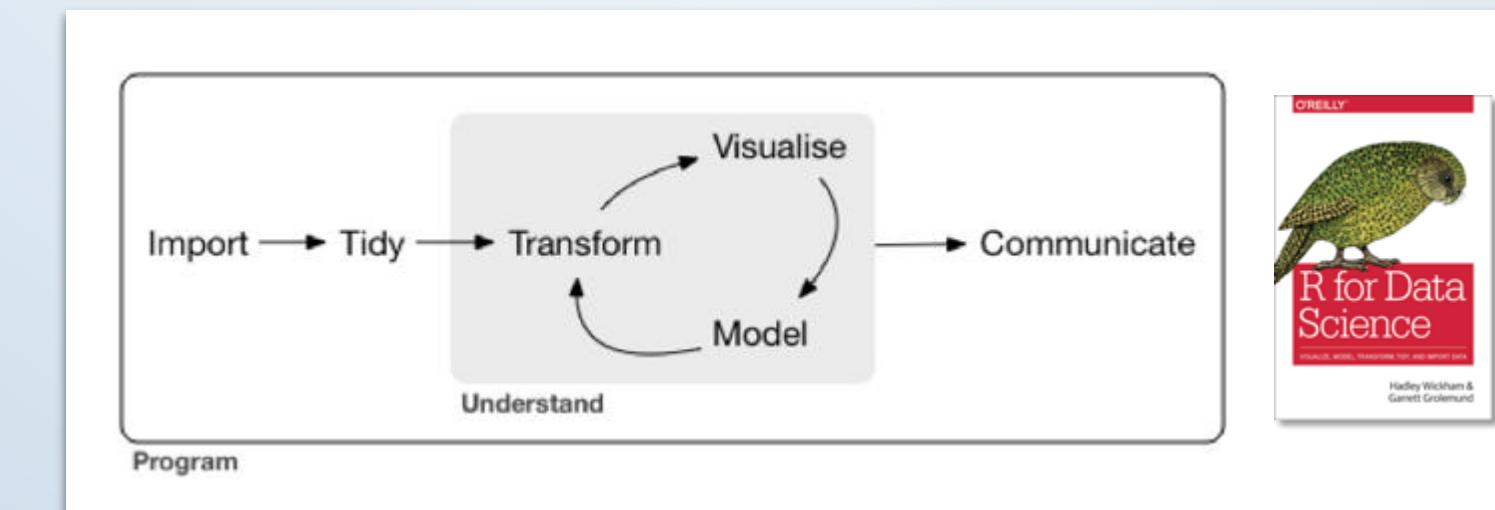
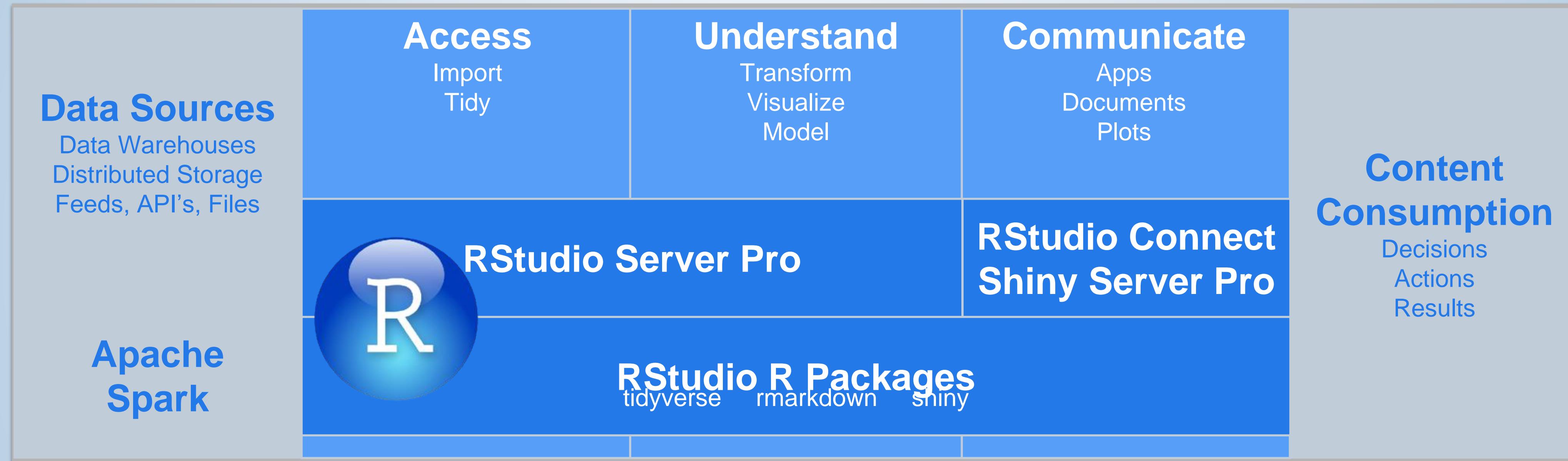


<http://r4ds.had.co.nz/intro.html>

HOW WE THINK ABOUT DATA SCIENCE



R for Data Science



The Mission of Rstudio is...

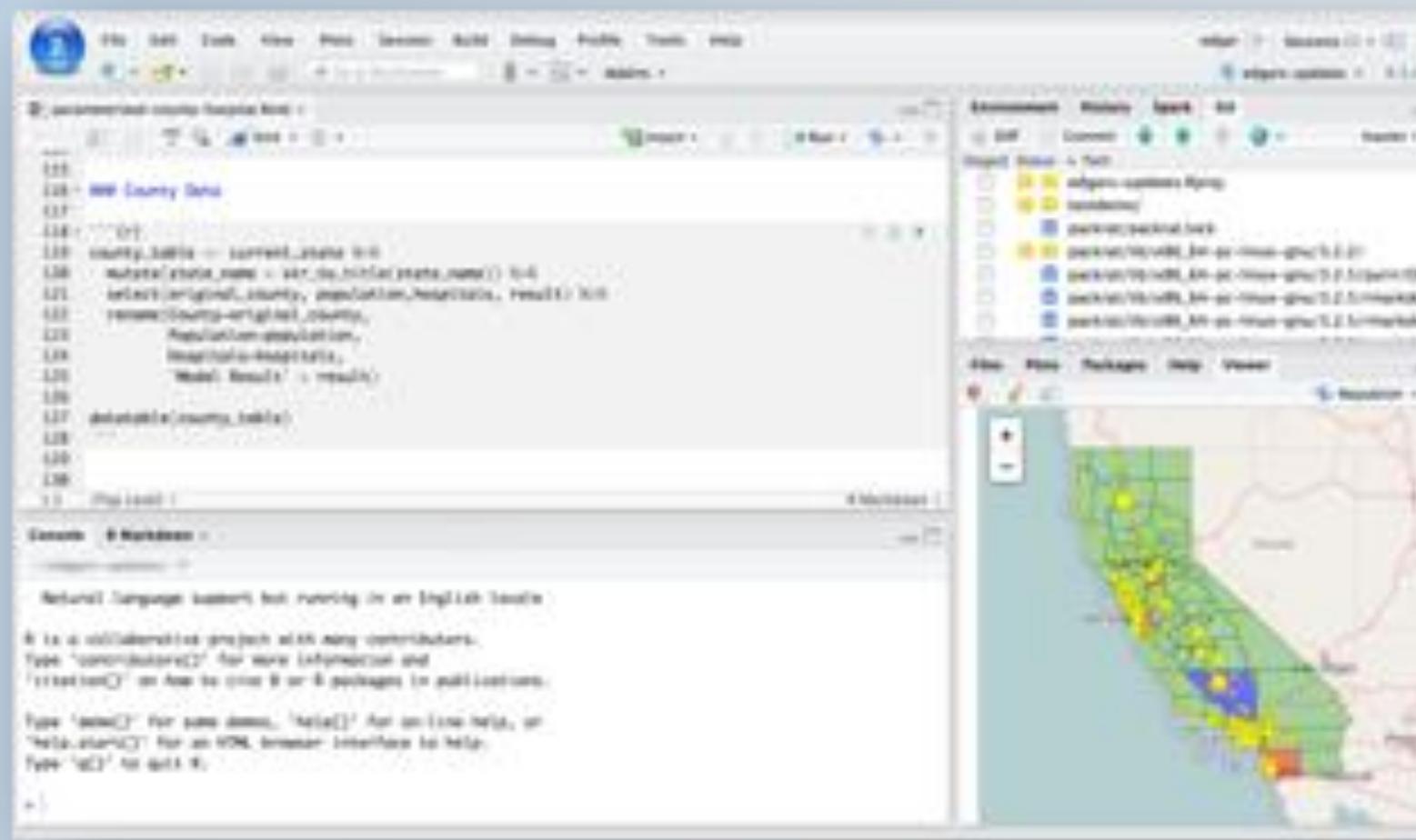
to provide the most widely used open source and enterprise-ready professional software for the R statistical computing environment.



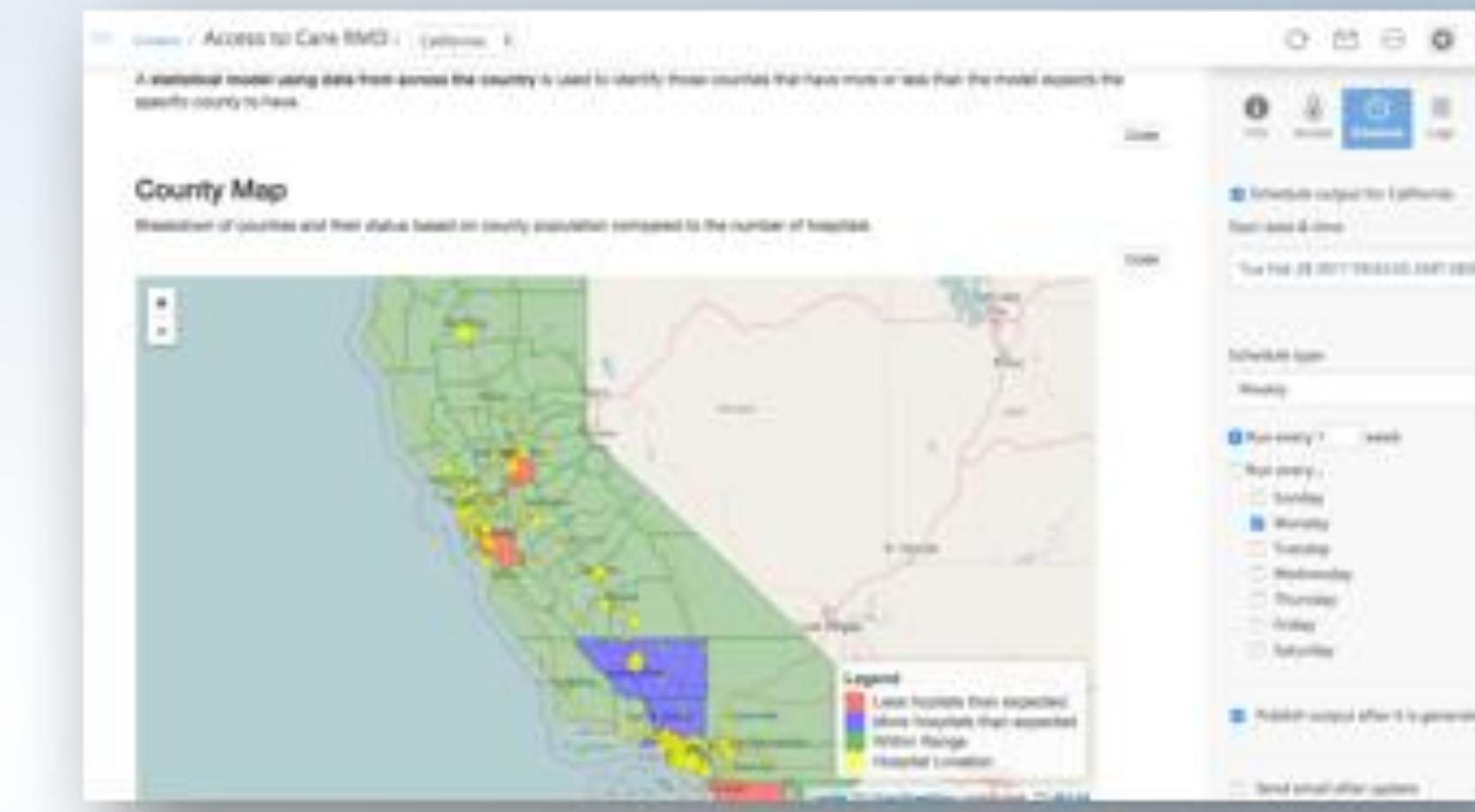
These tools further the cause of equipping everyone, regardless of means, to participate in a global economy that increasingly rewards **data literacy**.



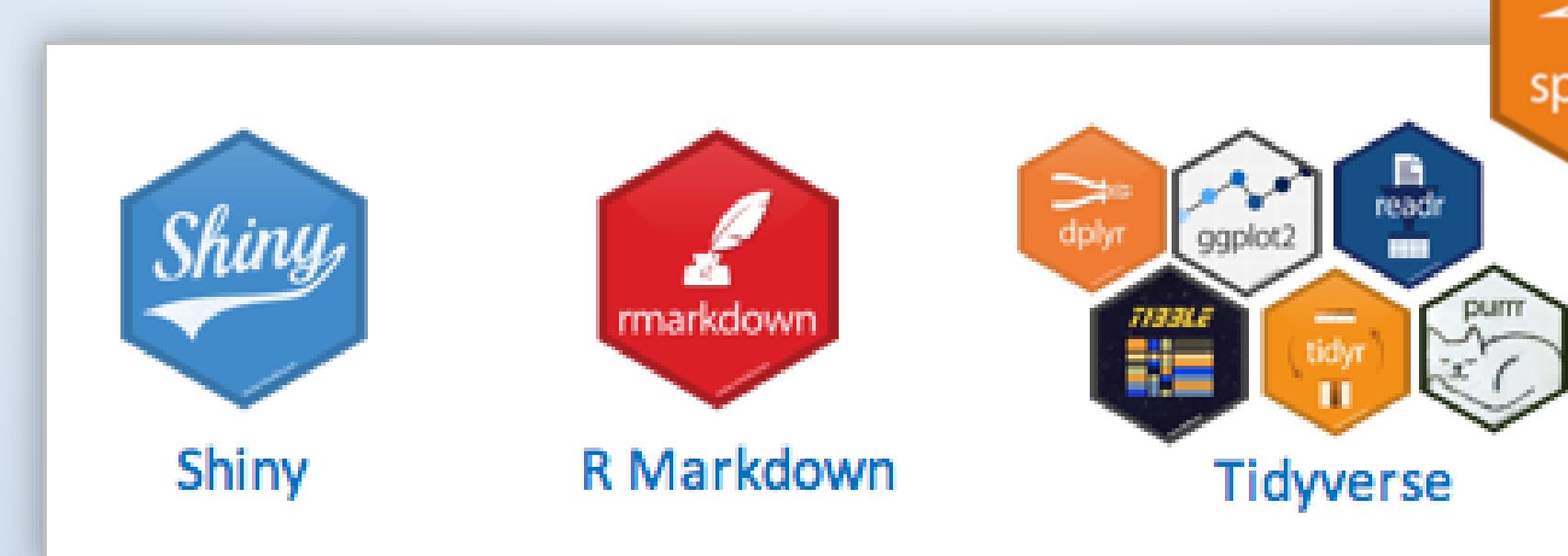
RStudio Products and Packages



RStudio Server Pro



RStudio Connect
Shiny Server Pro



R Packages



Introduction to Shiny

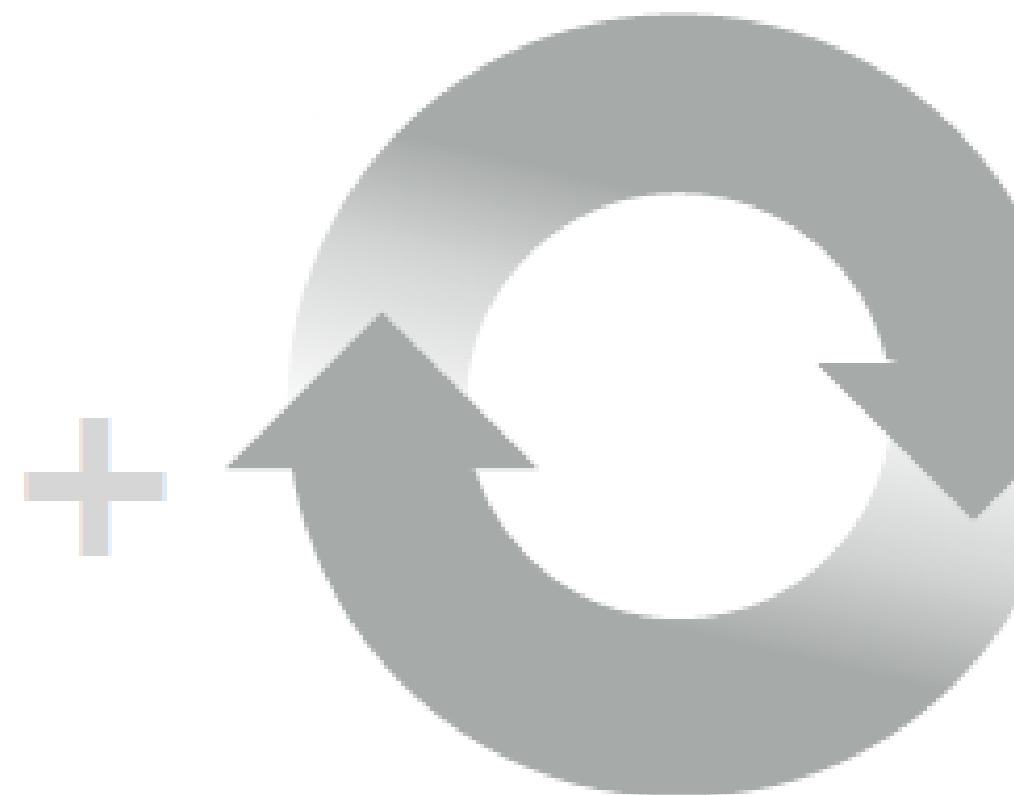
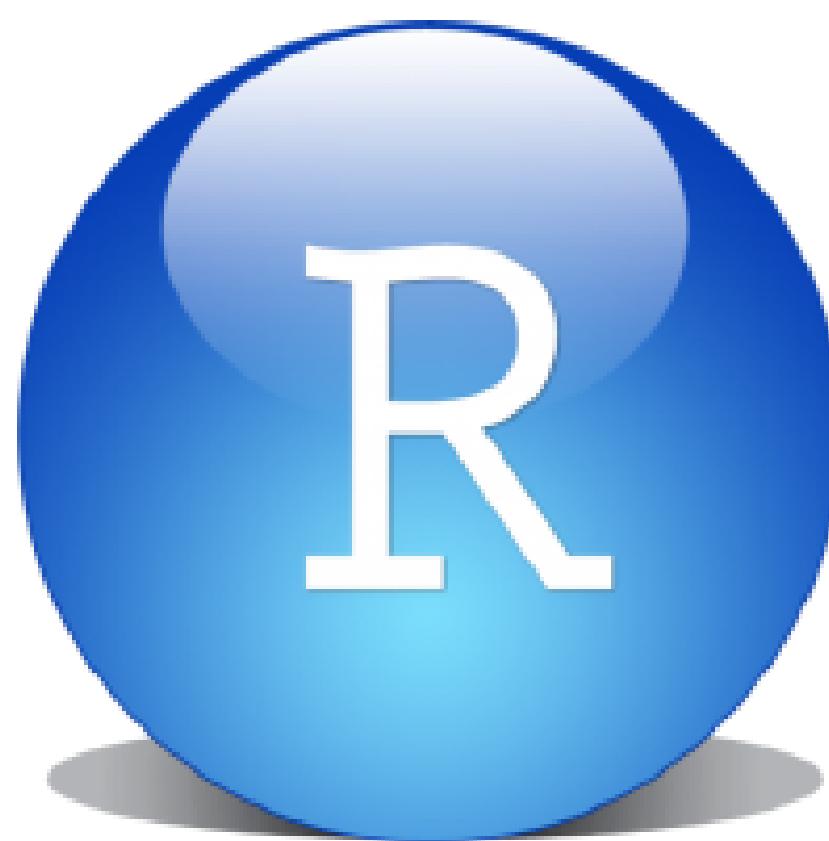


Motivation

- R is a powerful platform for data analysis
- State of the art statistical power
- A massive set of packages for statistical modelling, machine learning, visualization, and importing and manipulating data
- Large and enthusiastic community



Shiny



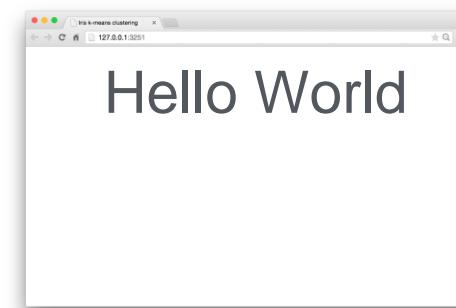
Reactive
Programming



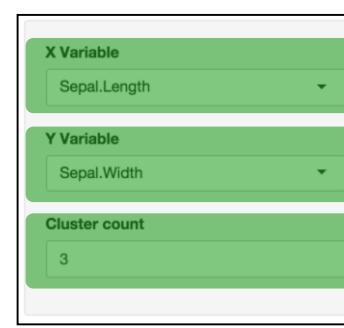
Web based
User Interface

Recap: UI

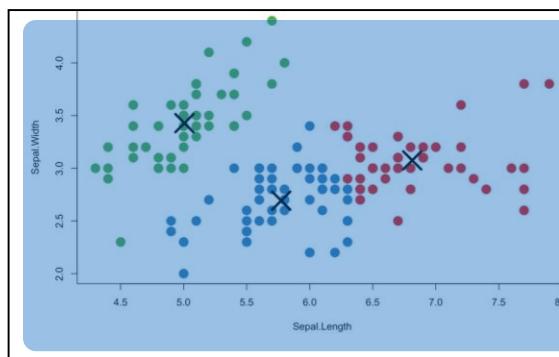
```
library(shiny)
ui <- fluidPage()
server <- function(input, output) {}
shinyApp(ui = ui, server = server)
```



Begin each app with the template



Add elements as arguments to **fluidPage()**



Create reactive inputs with an ***Input()** function

Display reactive results with an ***Output()** function

Recap: Server



`output$hist <-`

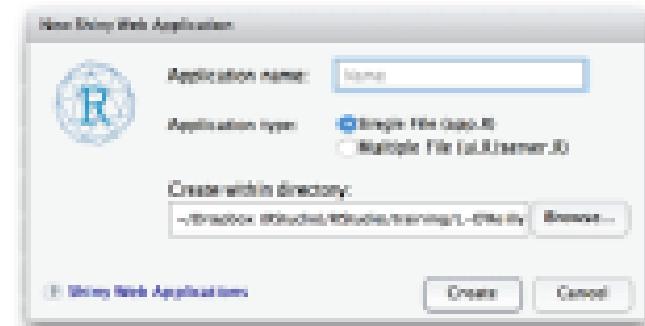
```
renderPlot({  
  hist(rnorm(input$num))  
})
```

`input$num`

Use the `server` function to assemble inputs into outputs. Follow 3 rules:

1. Save the output that you build to `output$`
2. Build the output with a `render*`() function
3. Access input values with `input$`

Recap



Open a new Shiny app with
File ▶ New File ▶ Shiny Web App...



Launch the app by opening app.R and
clicking **Run App**



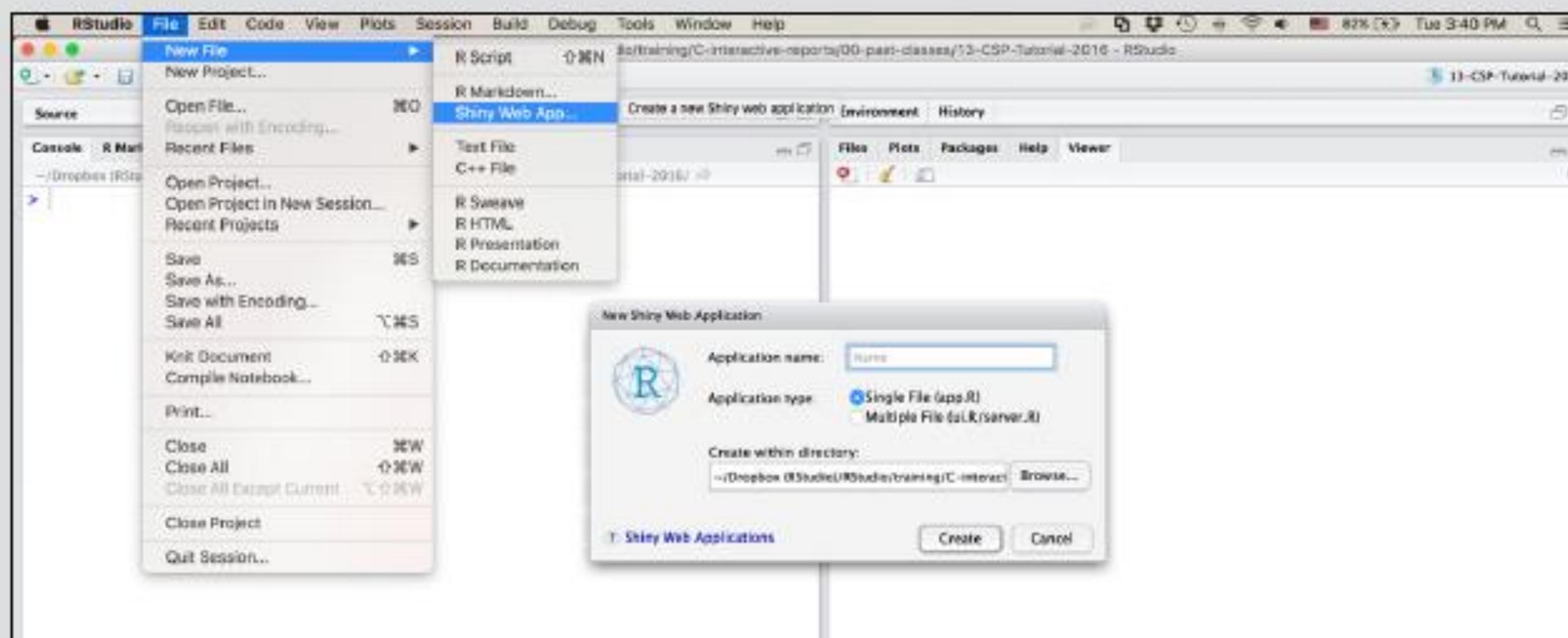
Close app by clicking the stop sign icon



Select view mode in the drop down
menu next to Run App

Warm up

1. Go to File ► New File ► Shiny Web App ► Single File
2. Click Run App at the top of the file to launch your first Shiny App.



03 : 00

© RStudio, Inc. All rights reserved.

runApp

You can launch any app from the command line
with `runApp`

```
runApp("~/Documents/App-1")
```

File path to app directory.
R will append the file path to the working directory,
if path does not begin at the home directory

What is the difference between Shiny and Shiny Server?

Shiny

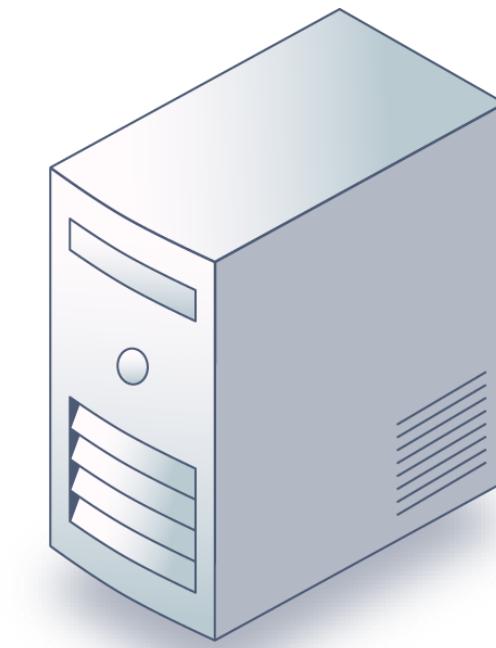


Free and open source R Package

Makes it incredibly easy to build interactive web applications with R

Automatic reactive binding between inputs and outputs and extensive pre-built widgets make it possible to build beautiful, responsive, and powerful applications.

Shiny Server



Software you install on your server

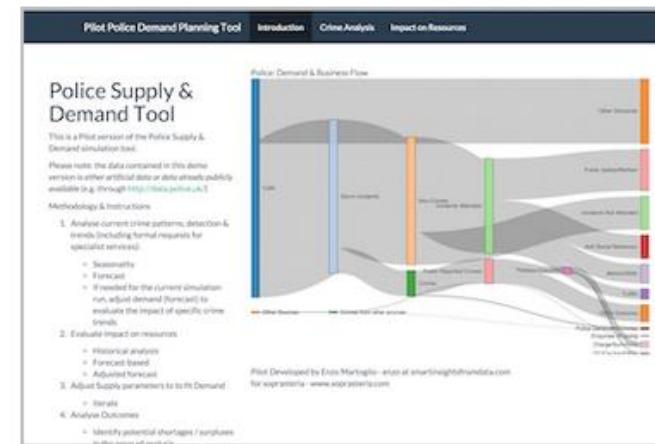
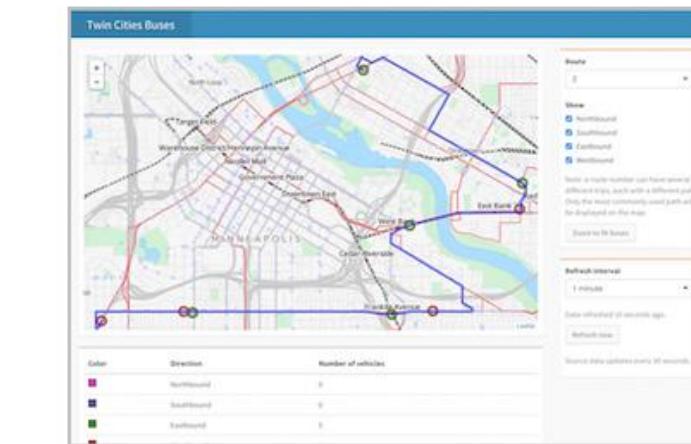
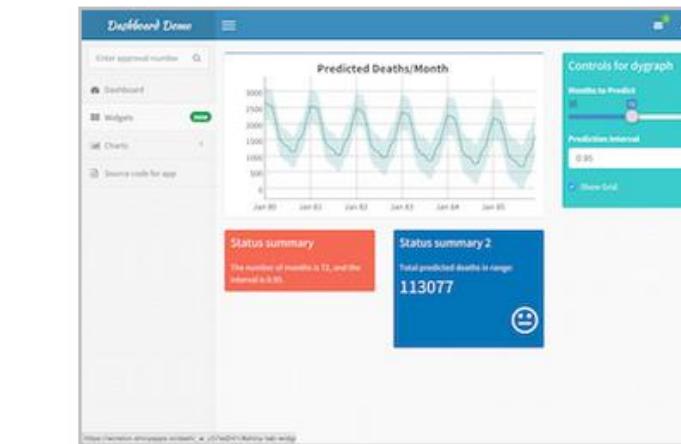
Enable users to host and managing Shiny applications

Scale a Shiny application to support many users
Protect and secure your applications
Manage the user experience

Shiny Showcase

www.rstudio.com/products/shiny/shiny-user-showcase/

Shiny Apps for the Enterprise



Shiny Dashboard Demo

A dashboard built with Shiny.

Location tracker

Track locations over time with streaming data.

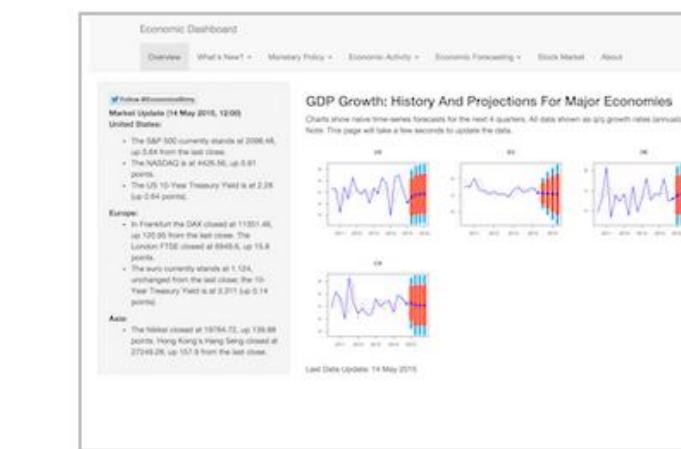
Download monitor

Streaming download rates visualized as a bubble chart.

Supply and Demand

Forecast demand to plan resource allocation.

Industry Specific Shiny Apps



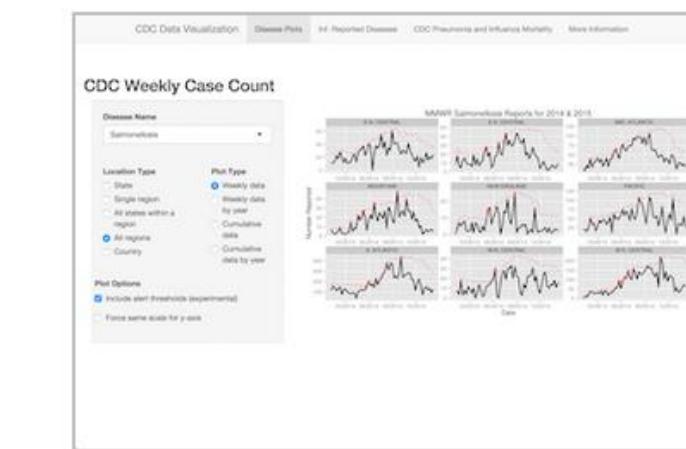
Economic Dashboard

Economic forecasting with macroeconomic indicators.



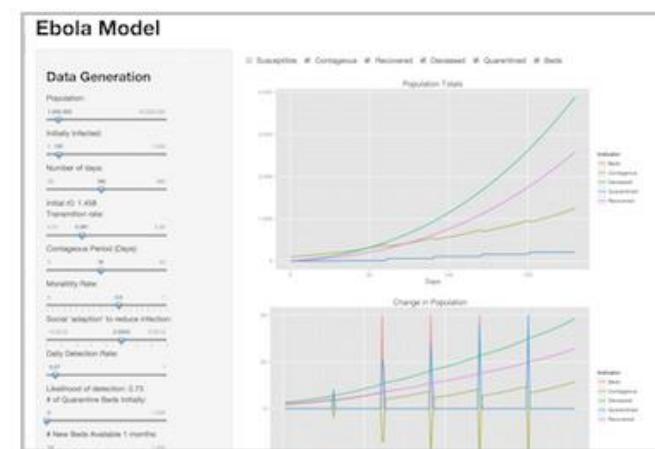
ER Optimization

An app that models patient flow.



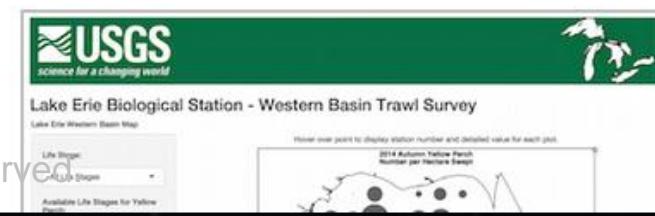
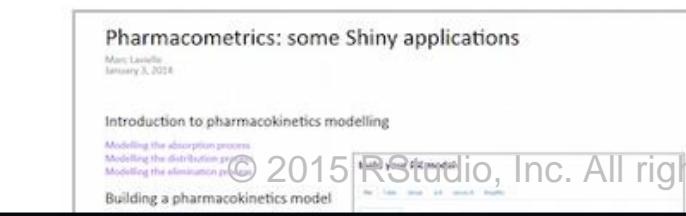
CDC Disease Monitor

Alert thresholds and automatic weekly updates.

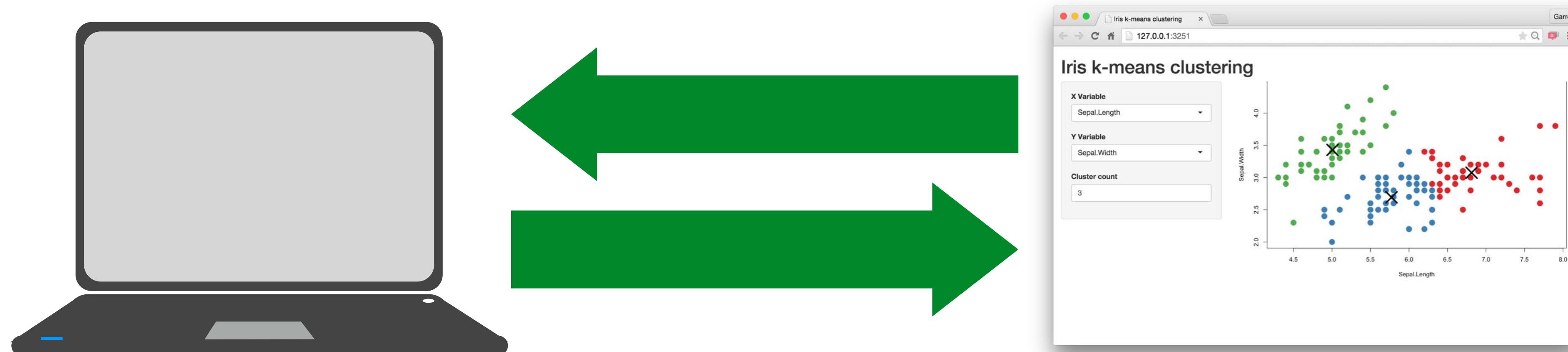


Ebola Model

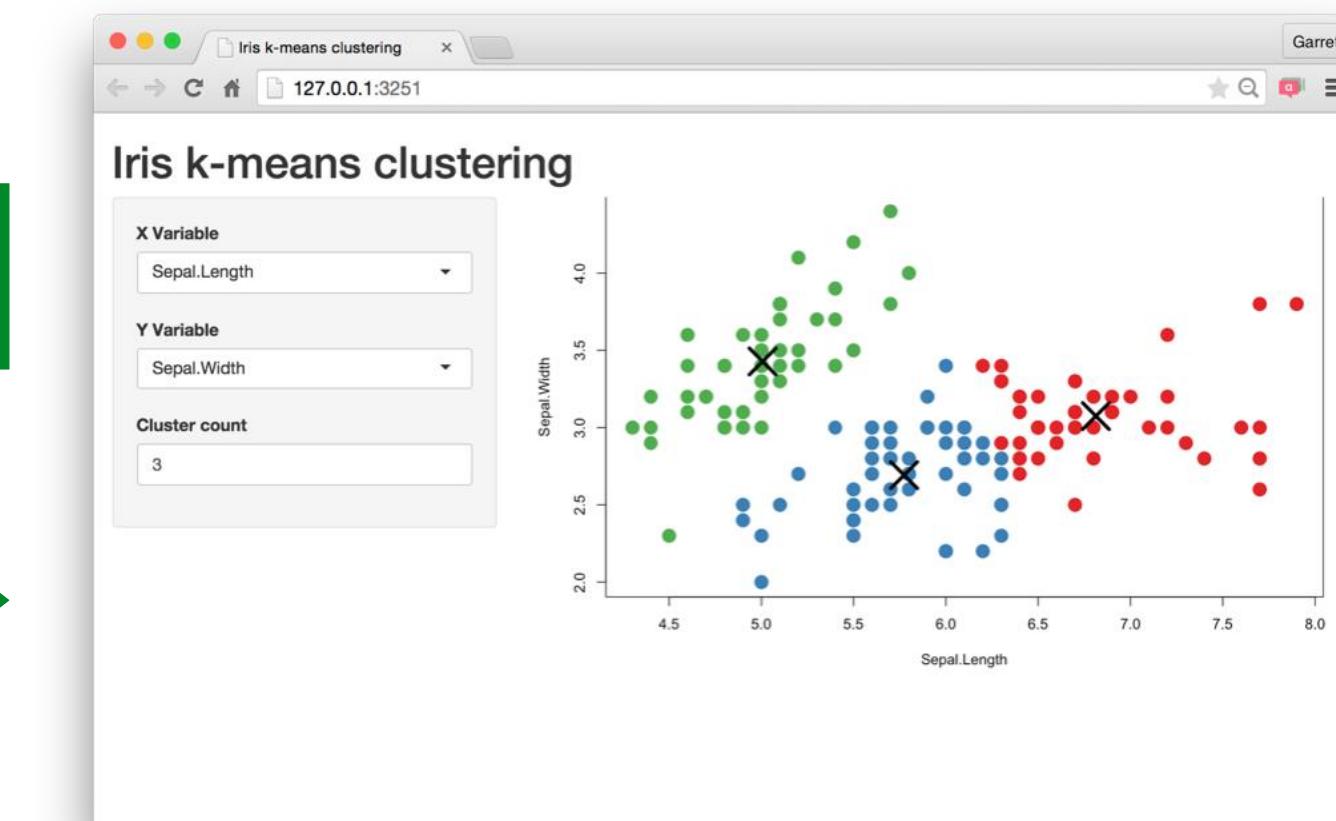
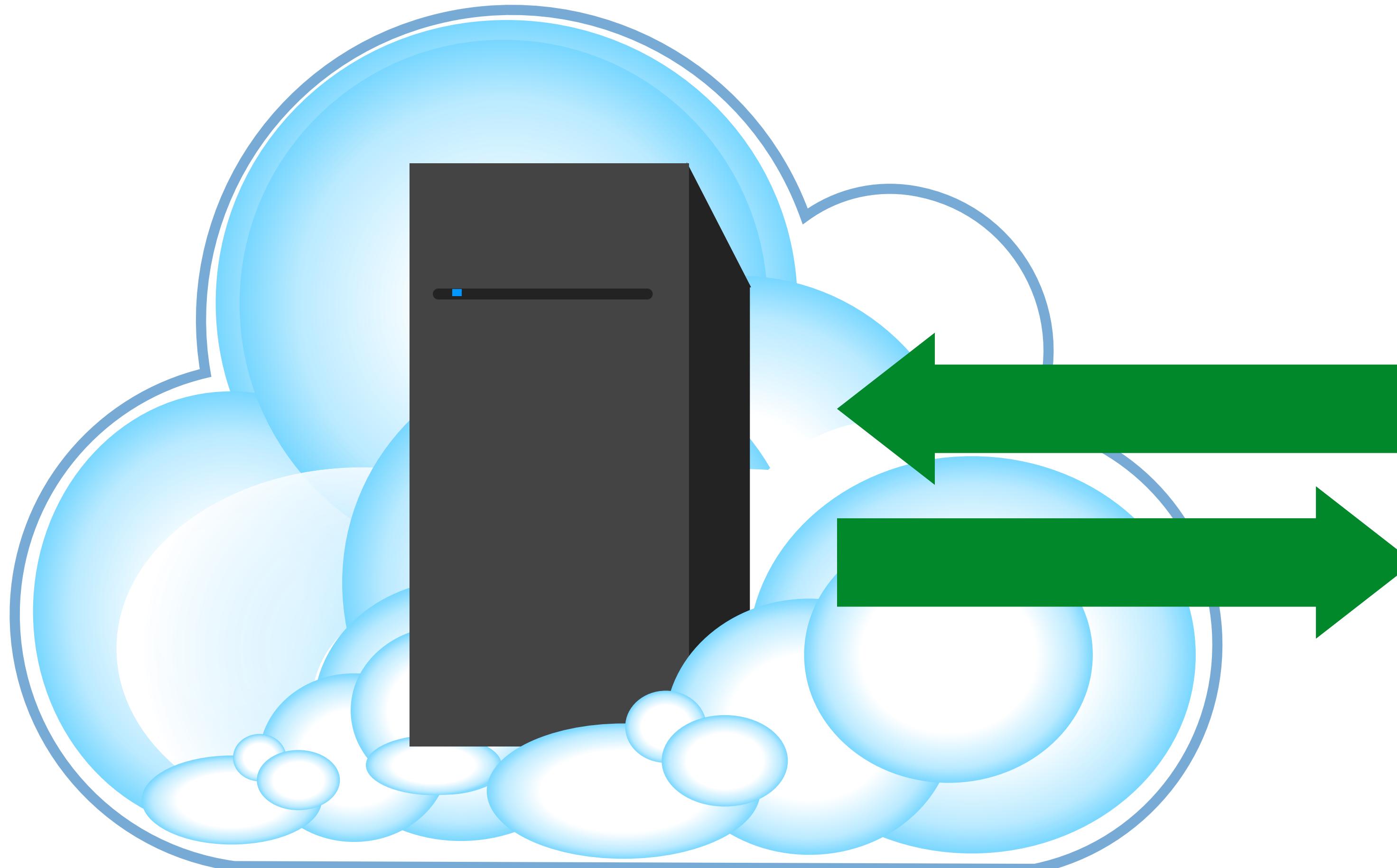
An epidemiological simulation.



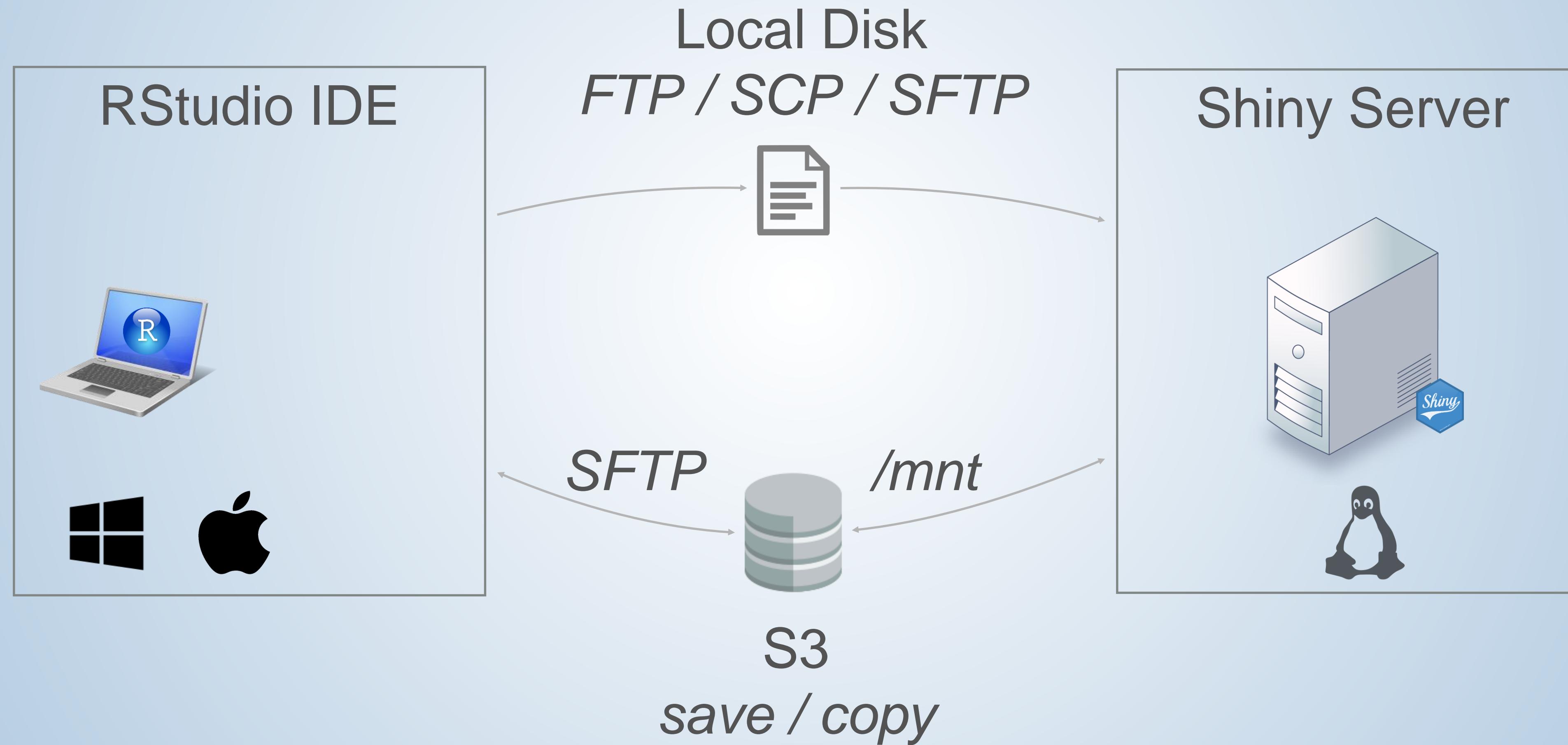
Every Shiny app is maintained by a computer running R



Every Shiny app is maintained by a computer running R



How do I deploy apps?



Develop and deploy on the same machine

RStudio Server Pro
Shiny Server Pro



Deploy Apps, R Markdown, and HTML



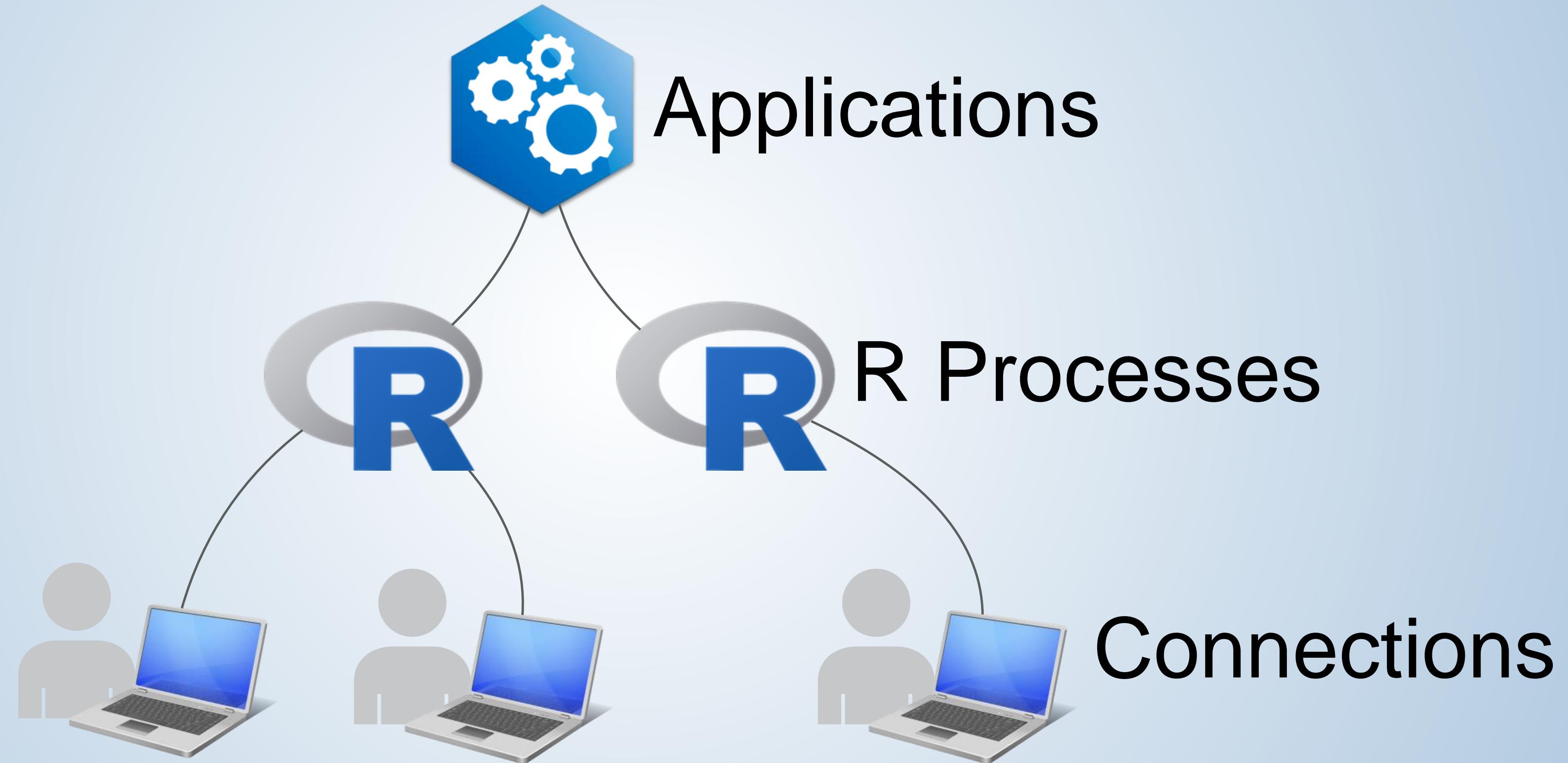
Shiny Server Pro

Shiny for the enterprise

Choose Shiny Server Pro to
secure user access, tune
application performance, monitor
resource utilization and get the
direct support you need to create
the best interactive data
experiences for your customers
and colleagues.



Scaling and Tuning Shiny Server Pro



Administrative Dashboards

Shiny Server Dashboard Applications Processes Connections admin ▾

Evaluation. Your trial license is set to expire in 6848 days.

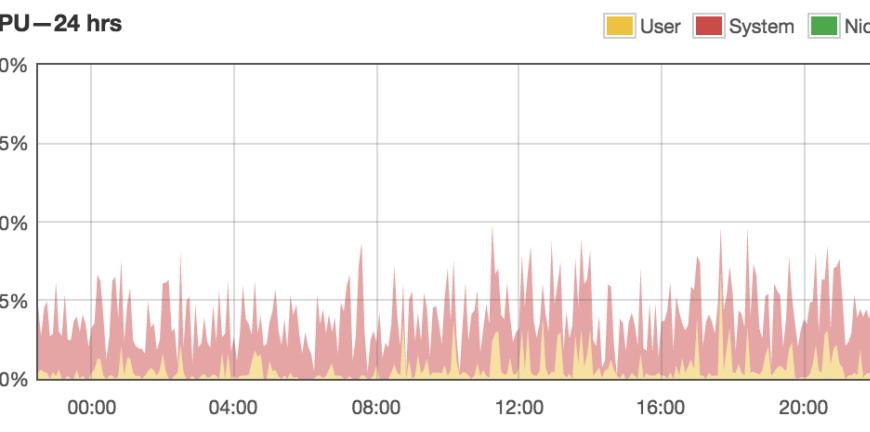
Load 

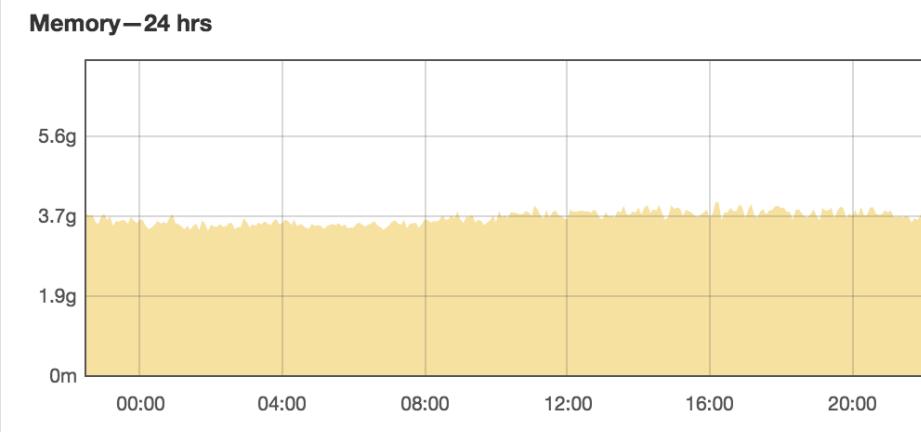
Memory 

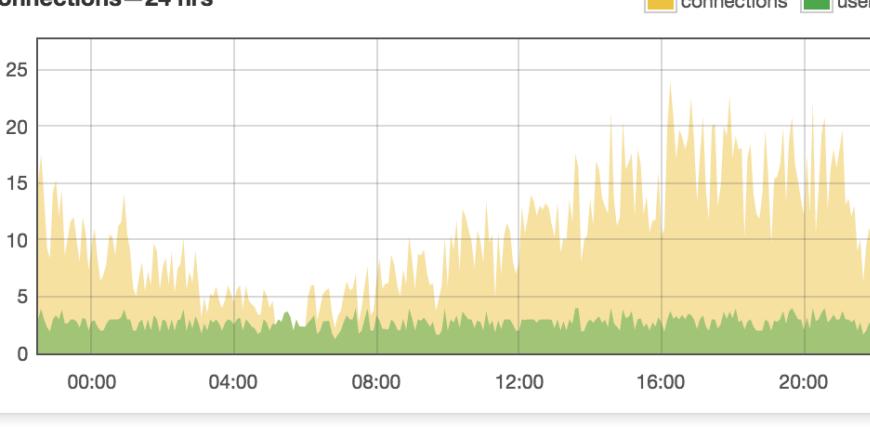
Active Users **2**

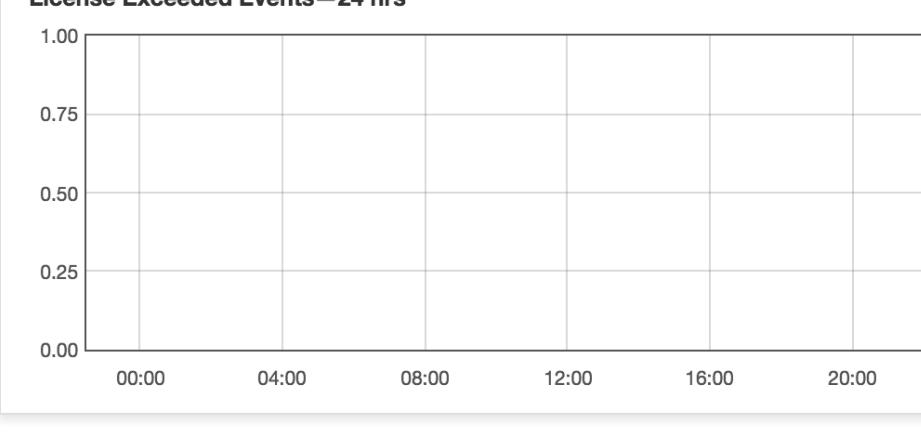
Open Connections **10**

System History

CPU—24 hrs 

Memory—24 hrs 

Connections—24 hrs 

License Exceeded Events—24 hrs 

System Information

Status	Authentication
License Status	Evaluation will expire in 6848 days.
Licensed Concurrent Users	Unlimited
Licensed Concurrent Connections	Unlimited
Running Shiny Applications	4
Running Shiny Processes	4

Shiny Server Dashboard Applications Processes Connections admin ▾

Evaluation. Your trial license is set to expire in 6848 days.

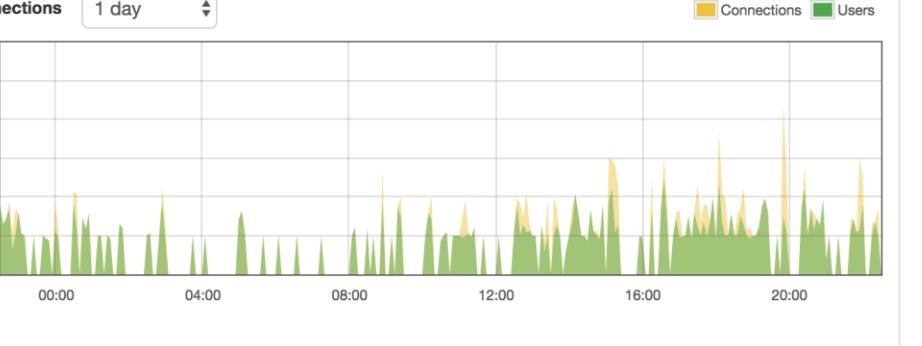
/srv/shiny-server/ssp-demo/07_widgets

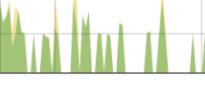
Current Status

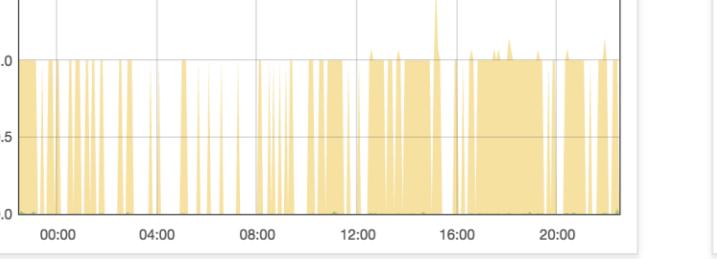
CPU Usage	0.00 of 1
Memory Usage	49MB
Current Processes	1
Busy Sessions	0
Active HTTP Requests	0
Recent Quota Exceeded Events	0

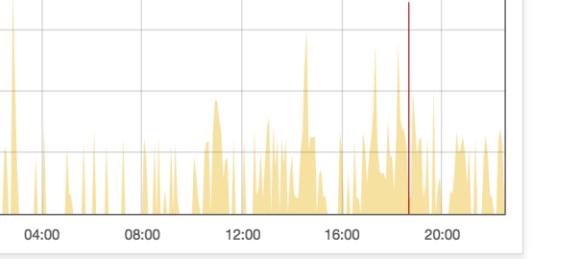
Open Connections **4**

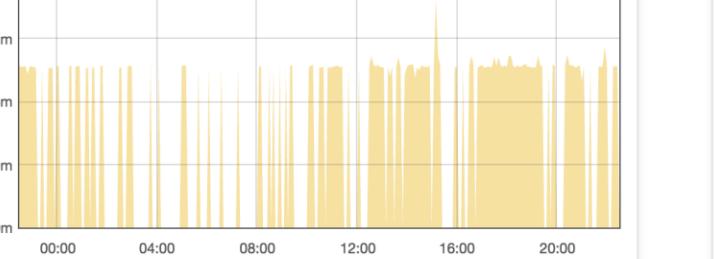
Active Users **2**

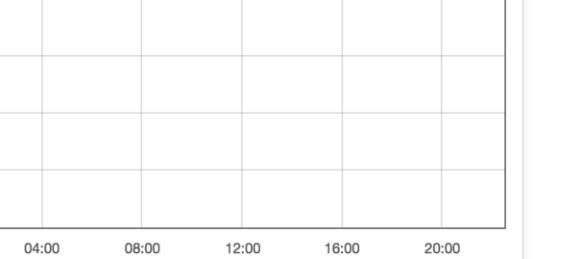
Connections 1 day 

Average Latency 

Processes 1 day 

Latency (ms) 1 day 

Memory 1 day 

Declined Requests/min 1 day 

Active Processes

Process	Application	Conn's	Latency	CPU	RAM
fd6be8b...	/srv/shiny-server/ssp-demo/07_widgets	4	28ms	0%	49MB

Active Connections

User	Initialized	Status
xhr-p	172.17.42.1	4 minutes ago
xhr-p	172.17.42.1	4 minutes ago
xhr-p	172.17.42.1	2 minutes ago
xhr-p	172.17.42.1	2 minutes ago

RSTUDIO CONNECT

A new publishing platform for
the work your teams create in
R.

Share Shiny applications, R
Markdown reports,
dashboards, plots, and more
in one convenient place.



RStudio Connect

A publishing platform for all the work your team creates in R

Content
Creators



Publish

Push button

RStudio Connect



Create variants
Schedule jobs
Distribute reports

Self-service



Content
Consumers



Make decisions
Take actions
See results

Connect is Built for Data Scientists and IT

Analyst

IT

Features

Push Button Deployment

Web UI for Managing Content

Email Integration

Job Scheduler

User Roles

Authentication: AD / PAM / LDAP

Metrics and Logging

Solutions

Rapid Iteration

Self Service

Content Sharing

Automatic Updates

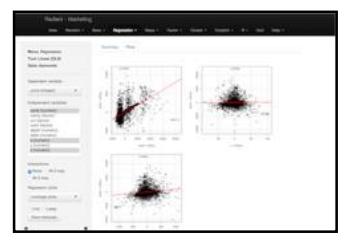
Access Control

Secure & Integrated

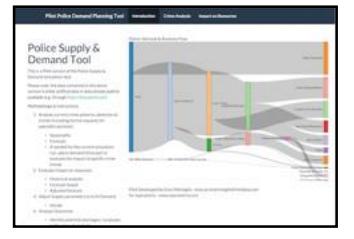
Scalable & Reliable

What can an
app do?

What can a Shiny app do?



Make R analysis accessible to **non-programmers**



Highly **customizable**, highly **shareable** HTML front end



Read and write to **databases**



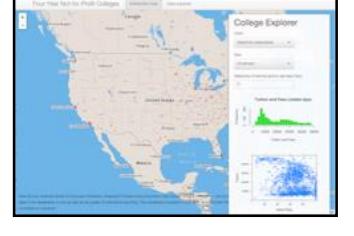
Monitor **streaming data**



Require and use **authentication**



Ideal for Exploratory Data Analysis



Ideal Data Portal / Results Explorer / Simulation API / Dashboard

Learn more

shiny.rstudio.com/articles

Interactive plots

Create interactive plots with base and ggplot2 graphics

- [Interactive plots](#)
- [Selecting rows of data](#)
- [Interactive plots - advanced](#)

The screenshot shows the Shiny website's "Articles" section. The left sidebar has a navigation menu with links to Overview, Tutorial, Articles (which is selected), Gallery, Reference, Deploy, and Help. The main content area is divided into several sections:

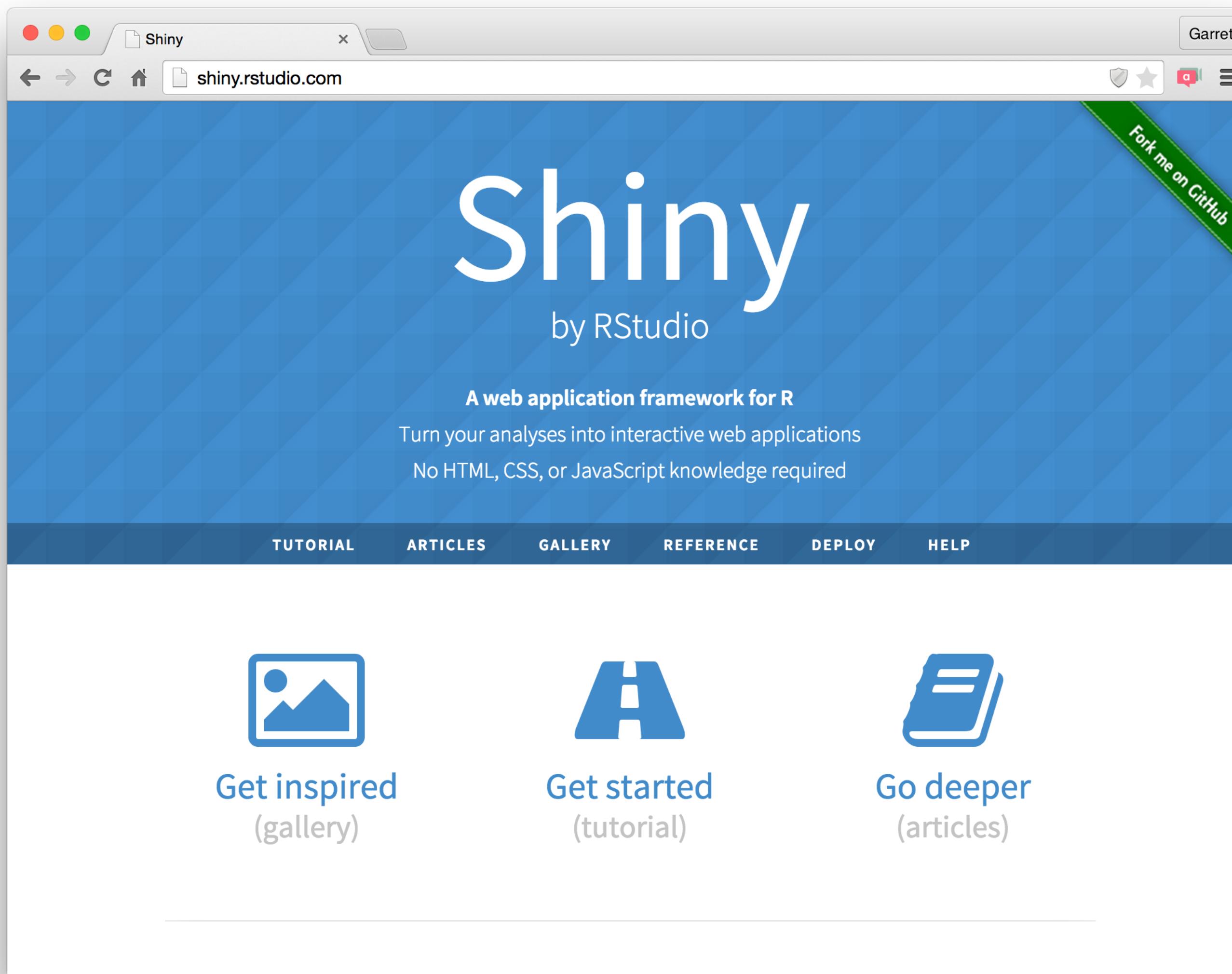
- Articles**: Sub-links include The basics, Extend Shiny, and a large list of other articles.
- Layouts and UI**: Sub-links include Application layout guide, Display modes, Tabs, Customize your UI with HTML, Build your entire UI with HTML, Build a dynamic UI that reacts to user input, Shiny HTML Tags Glossary, and Progress indicators.
- Interactive documents**: Sub-links include Introduction to R Markdown, Introduction to interactive documents, R Markdown integration in the RStudio IDE, and The R Markdown Cheat sheet.
- Outputs**: Sub-links include Render images in a Shiny app, How to use DataTables in a Shiny App, and a large list of other output-related articles.
- Best practices**: Sub-links include Write error messages for your UI with validate, Scoping rules for Shiny apps, Debugging techniques for Shiny apps, Learn about your user with session\$clientData, and Unicode characters in Shiny apps.
- Shiny Server Pro**: Sub-links include How to create User Privileges and Allow different libraries for different apps.
- Upgrade notes**: Sub-links include Notes for upgrading to particular versions of Shiny, Upgrade notes for Shiny 0.11, and Upgrade notes for Shiny 0.12.

At the bottom right, there is a small note: "Shiny is an RStudio project. © 2014 RStudio, Inc."

Teach yourself
Shiny

The Shiny Development Center

shiny.rstudio.com



The Shiny Cheat Sheet

www.rstudio.com/resources/cheatsheets/

Interactive Web Apps
with shiny Cheat Sheet
learn more at shiny.rstudio.com



Basics

A **Shiny** app is a web page (**UI**) connected to a computer running a live R session (**Server**)



Users can manipulate the UI, which will cause the server to update the UI's displays (by running R code).

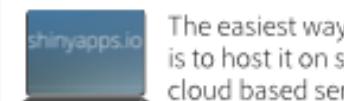
App template

Begin writing a new app with this template. Preview the app by running the code at the R command line.

```
library(shiny)
ui <- fluidPage()
server <- function(input, output) {
  shinyApp(ui = ui, server = server)
```

- ui** - nested R functions that assemble an HTML user interface for your app
- server** - a function with instructions on how to build and rebuild the R objects displayed in the UI
- shinyApp** - combines **ui** and **server** into a functioning app. Wrap with **runApp()** if calling from a sourced script or inside a function.

Share your app

shinyapps.io 

The easiest way to share your app is to host it on shinyapps.io, a cloud based service from RStudio

- Create a free or professional account at <http://shinyapps.io>
- Click the **Publish** icon in the RStudio IDE ($>=0.99$) or run:
`rsconnect::deployApp("<path to directory>")`

Build or purchase your own Shiny Server
 at www.rstudio.com/products/shiny-server/

Building an App - Complete the template by adding arguments to `fluidPage()` and a body to the `server` function.

Add inputs to the UI with `*Input()` functions
Add outputs with `*Output()` functions
Tell server how to render outputs with R in the server function. To do this:

- Refer to outputs with `output$<id>`
- Refer to inputs with `input$<id>`
- Wrap code in a `render*`() function before saving to output

```
library(shiny)
ui <- fluidPage(
  numericInput(inputId = "n",
    "Sample size", value = 25),
  plotOutput(outputId = "hist")
)
server <- function(input, output) {
  output$hist <- renderPlot({
    hist(rnorm(input$n))
  })
}
shinyApp(ui = ui, server = server)
```

Save your template as `app.R`. Alternatively, split your template into two files named `ui.R` and `server.R`.

```
library(shiny)
ui <- fluidPage(
  numericInput(inputId = "n",
    "Sample size", value = 25),
  plotOutput(outputId = "hist")
)
server <- function(input, output) {
  output$hist <- renderPlot({
    hist(rnorm(input$n))
  })
}
shinyApp(ui = ui, server = server)
```

ui.R
fluidPage(
 numericInput(inputId = "n",
 "Sample size", value = 25),
 plotOutput(outputId = "hist")
)

server.R
function(input, output) {
 output\$hist <- renderPlot({
 hist(rnorm(input\$n))
 })
}

ui.R contains everything you would save to ui.
server.R ends with the function you would save to server.

No need to call `shinyApp()`.

Inputs - collect values from the user

Access the current value of an input object with `input $<inputId>`. Input values are `reactive`.

Action `ActionButton(inputId, label, icon, ...)`

Link `actionLink(inputId, label, icon, ...)`

Choice 1 `checkboxGroupInput(inputId, label, choices, selected, inline)`

Choice 2 `checkboxInput(inputId, label, value)`

Choice 3 `checkboxInput(inputId, label, value)`

Check me `checkboxInput(inputId, label, value)`

Date `dateInput(inputId, label, value, min, max, format, startview, weekstart, language)`

Date range `dateRangeInput(inputId, label, start, end, min, max, format, startview, weekstart, language, separator)`

File `fileInput(inputId, label, multiple, accept)`

Numeric `numericInput(inputId, label, value, min, max, step)`

Text `passwordInput(inputId, label, value)`

Radio `radioButtons(inputId, label, choices, selected, inline)`

Select `selectInput(inputId, label, choices, selected, multiple, selectize, width, size) (also selectizeInput())`

Slider `sliderInput(inputId, label, min, max, value, step, round, format, locale, ticks, animate, width, sep, pre, post)`

Submit `submitButton(text, icon) (Prevents reactions across entire app)`

Text `textInput(inputId, label, value)`

RStudio® is a trademark of RStudio, Inc. • CC BY RStudio • info@rstudio.com • 844-448-1212 • rstudio.com • More cheat sheets at <http://www.rstudio.com/resources/cheatsheets/>

Learn more at shiny.rstudio.com/tutorial • shiny 0.12.0 • Updated: 6/15

© 2015 RStudio, Inc. All rights reserved.