

Case Studies in Customer Success

```
rstudio::conf_introduction(  
  presenter.name = "Katie Masiello",  
  presenter.company = "RStudio",  
  presenter.role = "Customer Success",  
  presenter.email = "katie.masiello@rstudio.com"  
)
```



kmasiello



@katieontheridge

rstudio::conf

Getting to Know RStudio Customers

{ Life Sciences Finance Manufacturing Insurance Healthcare Environmental Consulting Media
Education Nonprofit Technology Chemicals Public Sector Transportation Retail Energy } \subseteq Customers



REDFIN

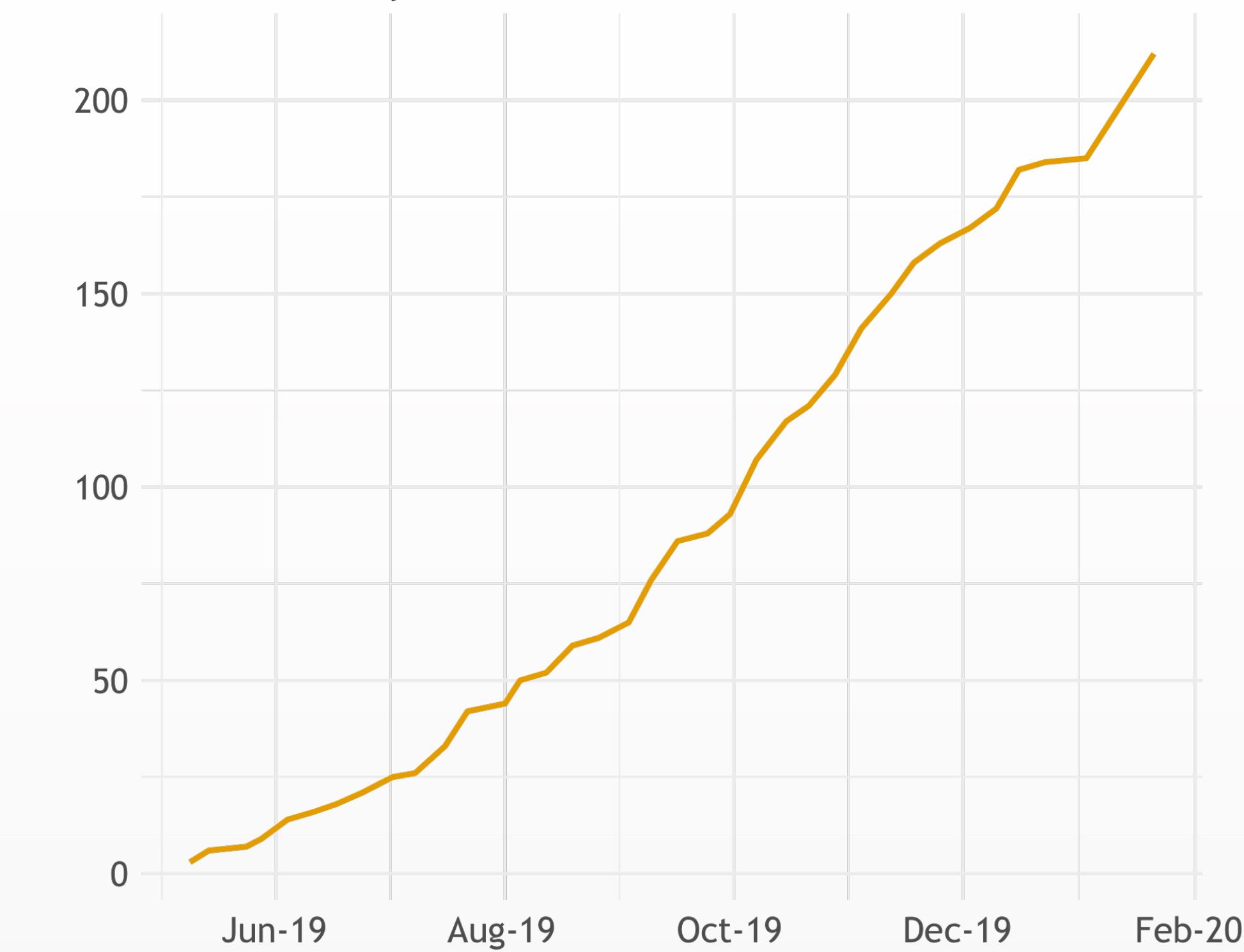


AVON



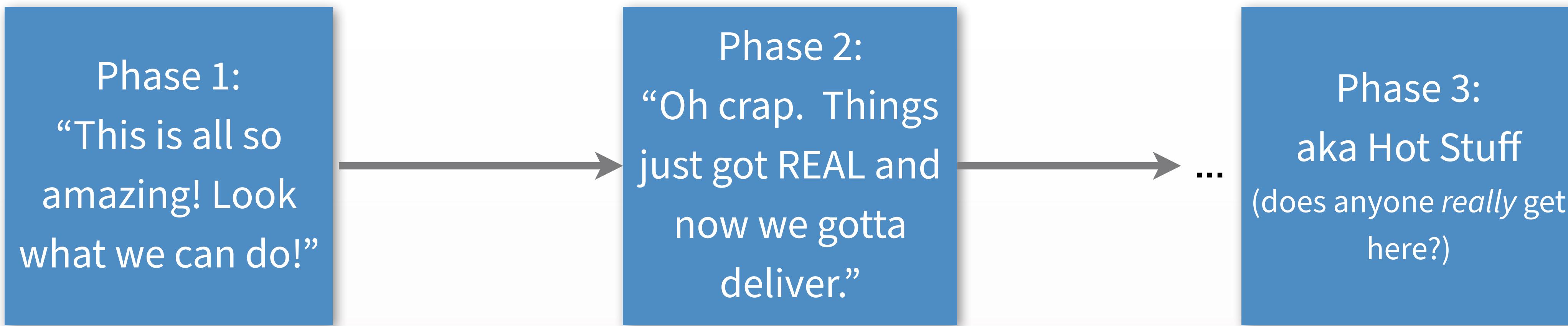
AMGEN

Count of Customer Conversations
Katie's history at RStudio



rstudio::conf

Phases of Data Science Team Maturity

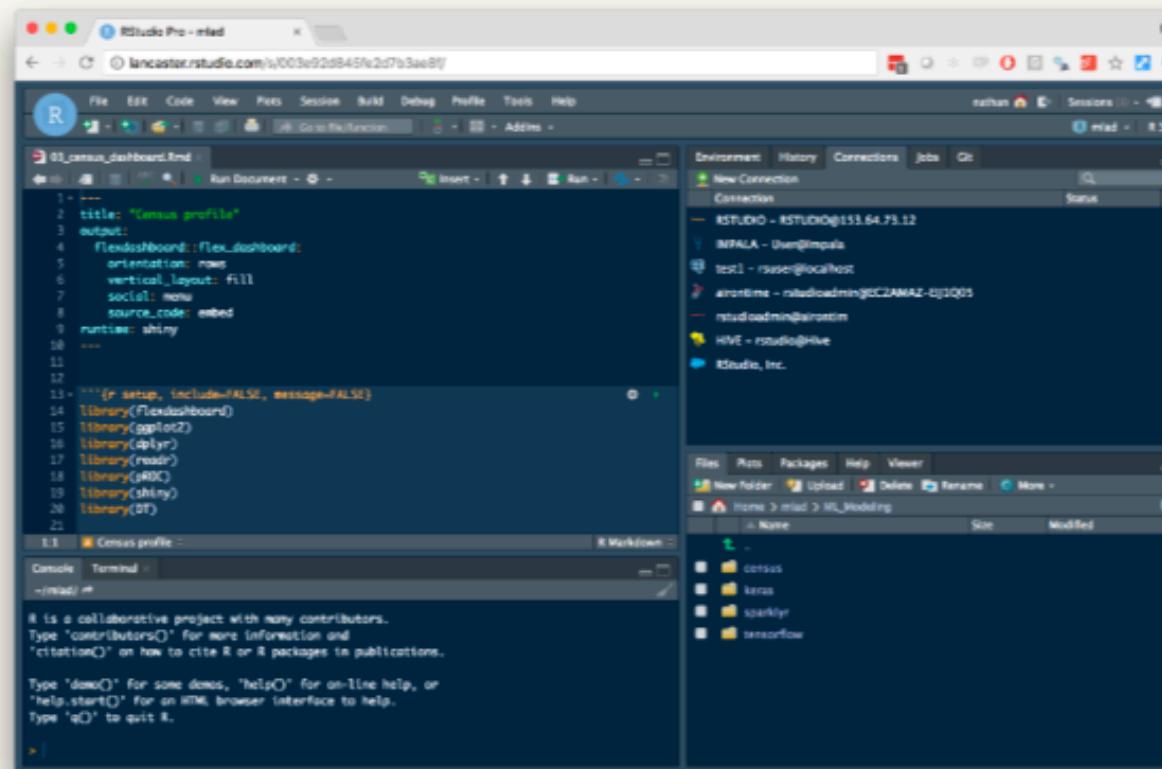


Data Science in the Enterprise

Data Sources

Data Science Lab

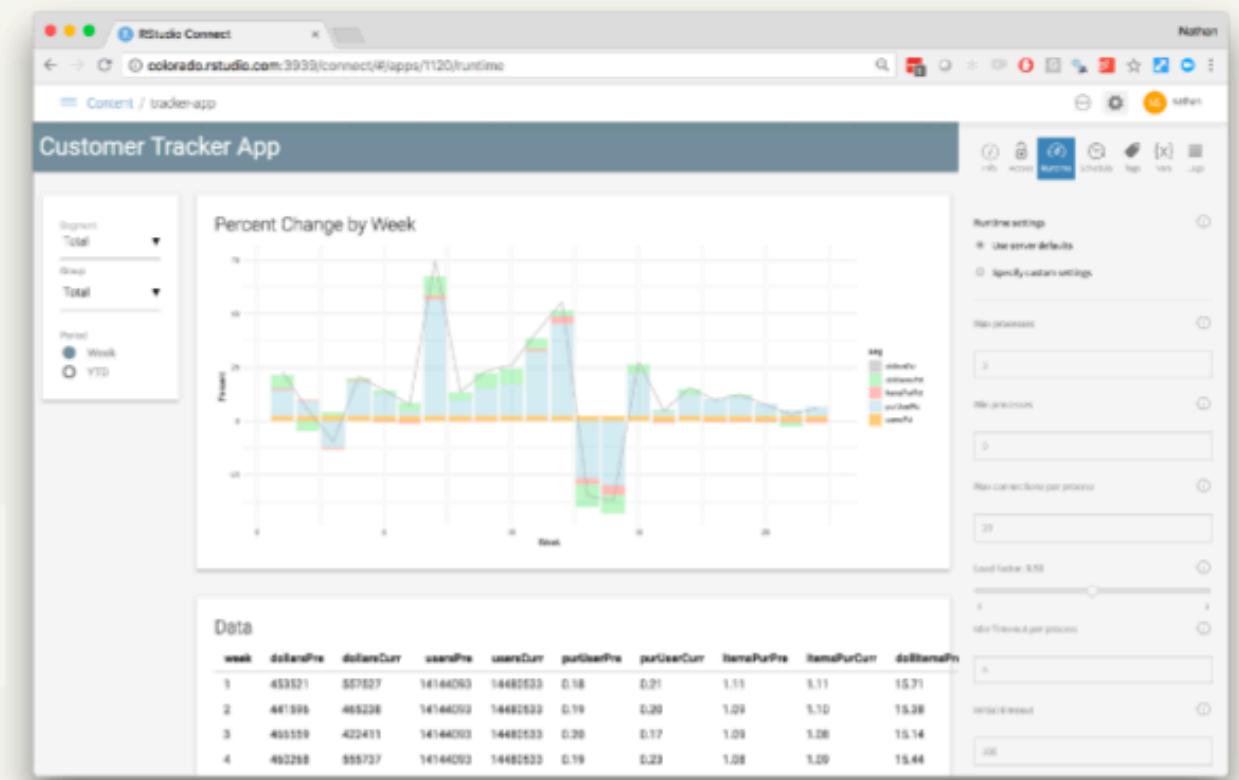
RStudio Server Pro Build Artifacts



RStudio Package Manager Manage Dependencies



RStudio Connect Run Artifacts



BI

In-Database

Reporting

Web

Workflows

API's

Programming Languages Operating Systems

Case Study: Phase 1 Team

Background:

Developed Shiny app to replace and extend spreadsheet-based analysis. Wide-eyed and excited about where they can go next.

Phase 1 Team in a Nutshell

Pain Points:

“Where do we put this data product so it can be used?”

“Can we get it there ourselves?”

Success is:

Having work visible and discoverable by stakeholders

Ultimate goal:

Secure buy in for their group’s analytical capabilities so they can expand their scope and impact by taking on more forward-thinking analytics.

Phase 1 Team: Success Strategy

RStudio Connect provides a home for data science assets with push-button deployment from RStudio or Jupyter IDEs

The screenshot illustrates the RStudio Connect deployment process. On the left, the RStudio interface shows an `app.R` script for a Shiny application. A callout arrow points from the `app.R` code to the **Publish to Server** dialog. The dialog shows the application title "Old Faithful" and the main panel UI code. A second callout arrow points from the **Publish** button in the dialog to the resulting RStudio Connect application on the right.

RStudio Connect Application:

- Content / shinyapp**
- Old Faithful Geyser Data**
- Number of bins:** A slider set to 30.
- Histogram of x**: A histogram showing the frequency distribution of the variable `x`.
- Who can view this application**: Specific users or groups: Kris Overholt, Tom Mock, Elena Ruiz.
- Who can change this application**: Katie Masiello, Kelly O'Briant.
- Who runs this content on the server**: The default user `rstudio-connect`.

Phase 1 Team: Enabled Growth

- Connect opened the door to tackle other inefficiencies in their process using the variety of data products supported on Connect



- Scheduled and ad hoc reporting with R Markdown and Jupyter Notebooks
 - Condition-triggered and customized emails using `blastula`
 - `pins` for ephemeral ETL data

Report Scheduling

Content / connect-example-blastula

Main Document

An R Markdown Document

This is an **R Markdown** document. Markdown is a simple formatting syntax for writing on the web. **R Markdown** takes this a step further by combining Markdown prose and R code into reproducible documents that can be output as HTML, PDF, Word, and many more output formats. For more details on using **R Markdown**, have a look through [its documentation site](#).

Here is some R code, which is contained within a code chunk:

```
diamonds %>%
  group_by(carat, cut) %>%
  summarize(mean_price = mean(price)) %>%
  filter(cut != "Fair", carat < 3) %>%
  ggplot() +
  geom_point(aes(x = carat, y = mean_price)) +
  stat_smooth(
    aes(x = carat, y = mean_price),
    method = "gam"
  ) +
  facet_wrap(facets = vars(cut)) +
  labs(
    title = "Diamond Prices",
    subtitle = "Faceted by Diamond Cut",
    caption = "Source: The [diamonds] dataset in {ggplot2}.",
    x = "Carats", y = "Mean Price, US Dollars"
  ) +
  scale_y_continuous(labels = scales::dollar)
```

Diamond Prices
Faceted by Diamond Cut

RStudio Connect:

Set the report execution schedule and email recipients list

KO kelly

Info Access Runtime Schedule Tags Vars Logs

Schedule output for default

Start date & time

Thu Dec 26 2019 12:29:49 GMT-0500

Schedule type

Monthly

Run every 1 month...

on the 1 st day of the month.

on the first Monday of the month.

Publish output after it is generated

Send email after update

rstudio::conf

ETL with R Markdown and pins

Get data from API

```
# Get list of feeds
feeds <- feeds_urls()

# Get station status Data and clean
dat <- feeds %>%
  filter(name == "station_information") %>%
  pull("url") %>%
  get_data() %>%
  extract2("data") %>%
  select(station_id, name, lat, lon) %>%
  unique()
```



Pin

```
pins::pin(dat,
  "bike_station_info",
  "List of stations of Capitol Bikeshare",
  "rsconnect")
```

```
## # A tibble: 578 x 4
##   station_id name      lat    lon
##   <chr>        <chr>    <dbl> <dbl>
## 1 1            Eads St & 15th St S 38.9 -77.1
## 2 2            18th St & S Eads St 38.9 -77.1
## 3 3            Crystal Dr & 20th St S 38.9 -77.0
## 4 4            Crystal Dr & 15th St S 38.9 -77.0
## 5 5            Aurora Hills Cmty Ctr / 18th St & S Hayes St 38.9 -77.1
## 6 6            Pentagon City Metro / 12th St & S Hayes St 38.9 -77.1
## 7 7            Army Navy Dr & S Joyce St 38.9 -77.1
## 8 8            Crystal City Metro / 18th St & S Bell St 38.9 -77.1
## 9 10           Crystal Dr & 27th St S 38.8 -77.1
## 10 11          S Glebe Rd & Potomac Ave 38.8 -77.1
## # ... with 568 more rows
```

Download Pin

[data.csv](#)

Use Pin from R Python

```
# Register RStudio Connect
library(pins)
board_register("rsconnect", server = "https://colorado.rstudio.com/rsc")

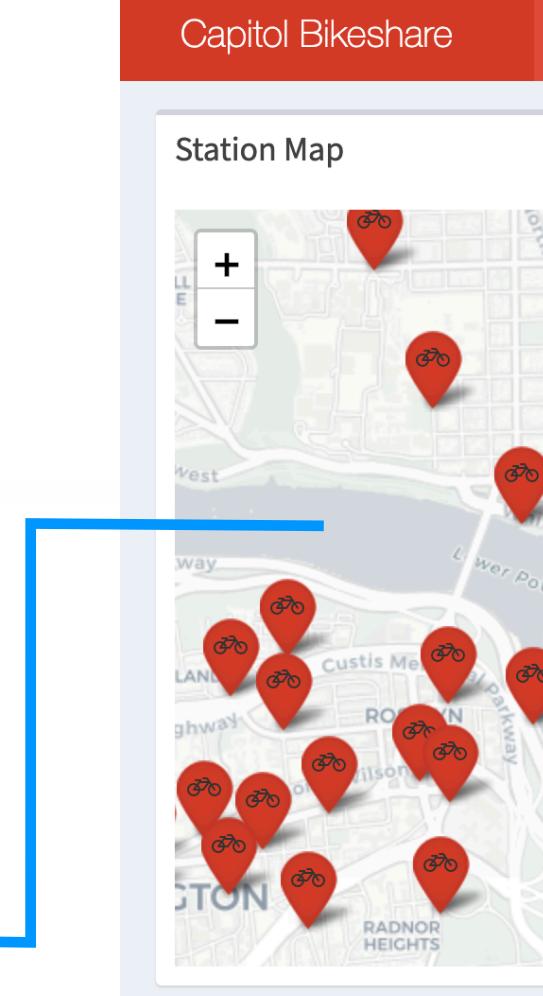
# Retrieve Pin
pin_get("alex.gold/bike_station_info", board = "rsconnect")
```



Preview Pin

station_id	name	lat	lon
1	Eads St & 15th St S	38.858971	-77.05323
2	18th St & S Eads St	38.85725	-77.0532
3	Crystal Dr & 20th St S	38.856425	-77.049232
4	Crystal Dr & 15th St S	38.86017	-77.049593
5	Aurora Hills Cmty Ctr / 18th St & S Hayes St	38.857866	-77.05949
6	Pentagon City Metro / 12th St & S Hayes St	38.862303	-77.059936
7	Army Navy Dr & S Joyce St	38.8637	-77.0633
8	Crystal City Metro / 18th St & S Bell St	38.857404962615	-77.0511317253113
10	Crystal Dr & 27th St S	38.848466	-77.051514
11	S Glebe Rd & Potomac Ave	38.8426	-77.0502
12	Crystal Dr & 23rd St S	38.8533	-77.0498
13	Clark St & 26th St S	38.850688	-77.05152
14	19th St & Pennsylvania Ave NW	38.9003	-77.0429
15	14th & V St NW	38.917931	-77.032112
16	11th & Kenyon St NW	38.929464	-77.027822
17	16th & Harvard St NW	38.926088	-77.036536
18	Adams Mill & Columbia Rd NW	38.922925	-77.042581
19	14th & Harvard St NW	38.9268	-77.0322

1-18 of 578 rows



rstudio::conf



Case Study: Phase 2 Team

Background:

Team with a growing portfolio of models and apps of increasing importance to the organization published to Connect. Burned a few times deploying straight to production. They know process rigor and version control is in their best interest but adoption is slow.

Phase 2 Team in a Nutshell

Pain Points:

- Disparate opinions regarding deployment strategies for publishing content (push button vs IT-controlled)
- Growing pains in adopting version control and process rigor
- Overall angst of what do we do now to keep this thing going?!

Success is:

Efficient but managed deployment workflow with a clear understanding of what “Production” means

Ultimate goal:

Collaborative relationship with IT with a foundation of best practices

What is “Production” anyway?



Incomplete Concept of “Production”

Incomplete mindset:

“My data product is accurate and to be used for informing business decisions.”

Famous last words:

“It works fine on my desktop, let’s go to Production!”



Formalized Production

- **Keep it correct:** works as intended, provides the right answers
- **Keep it up:** unplanned outage are rare or nonexistent
- **Keep it safe:** data, functionality, and code are all kept safe from unauthorized users
- **Keep it snappy:** fast response times, ability to predict needed capacity for expected traffic
- **State of mind:** expect that every change has the potential to disrupt; design and test for this to proactively address

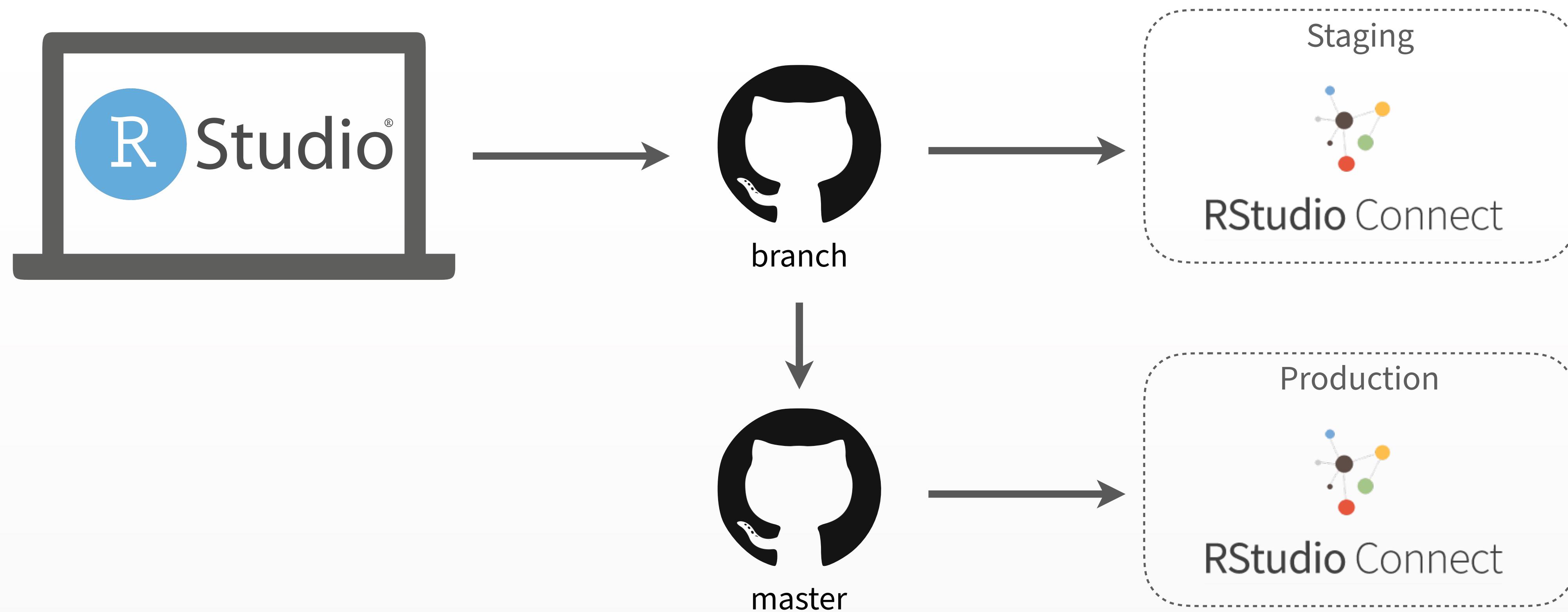


Phase 2 Team: Success Strategy

- RStudio Connect provides the “where” for a formalized production definition
 - Scalability and performance tuning
 - Security
 - Staging and Production environments
- Cultivating the Production mindset with implementation of dev-test-prod workflow
 - Efficient and managed solution with automated Git-backed deployment to Connect for Staging and Production

Automation in Dev-Test-Prod

Git-backed publishing automates deployment to Staging and Production environments



Demo: Automated Dev-Test-Prod with Git-backed Deployment

The screenshot shows the RStudio Connect interface with four listed environments:

- Production - git-backedShiny**: Type: Application, Deployed: Today at 3:35 PM from Git. Preview image: Old Faithful Geyser Data histogram.
- Staging - git-backedShiny**: Type: Application, Deployed: Today at 3:29 PM from Git. Preview image: Old Faithful Geyser Data histogram.
- dynamic-classroom**: Type: Application, Deployed: Today at 3:16 PM. Preview image: Workshop Servers welcome screen.
- Pro Admin Training 1-2 exercise: Install and configure RSP**: Type: Document, Deployed: Today at 1:40 PM. Preview image: RSP configuration interface.

A sidebar on the right contains options for publishing, visibility, technology, projects and presentations, and solutions.

Options

- Visible to You
- Content You Own
- All Server Content

Technology

- Deep Learning
- Output Types
- Email
- BackEnds
- Dashboards

Projects and Presentations

- Calendars
- QuickStart
- Production Webinar
- Pins Demo: Bikes
- Team Admin Training

Solutions

- Access to Care
- Python



Closing Thoughts

There are no data science unicorns



...teamwork makes the dream work



Wrapping Up

- Your success story emerges as you remove barriers.
 - Share your data products
 - Leverage automation and reproducibility to make efficient workflows
 - Know where you're going and that it takes a whole team to get there
- Think forward, ask questions, come talk with us at the Lounge

More Information



- Read Customers' stories - <https://rstudio.com/about/customer-stories/>
- Resources
 - RStudio Connect in Production - <https://resources.rstudio.com/webinars/rstudio-connect-in-production>
 - Publish Content from Git Repositories - <https://solutions.rstudio.com/demos/git-deployment/>
 - Diffable Data Science - <https://solutions.rstudio.com/2020/01/10/diffable-data-science-demo-with-bike-prediction/>
- Try RStudio Professional Products
 - RStudio Team QuickStart - <https://rstudio.com/products/quickstart/>
 - Evaluation options - <https://docs.rstudio.com/evaluation/>

