

Introducing flexdashboard

AN R MARKDOWN FORMAT FOR QUICK, INTELLIGENT DASHBOARDS

What is a flexdashboard?

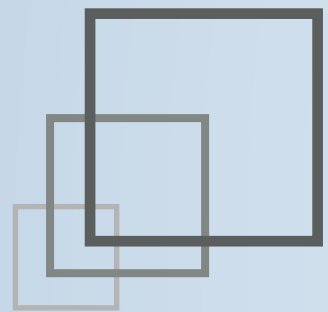
<http://rmarkdown.rstudio.com/flexdashboard/examples.html>

flexdashboards



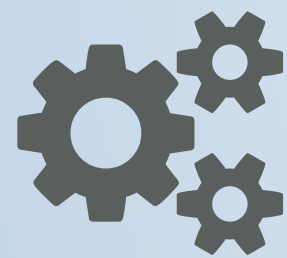
1. **Dashboards based on R Markdown**

Easy to compose, easy to publish



2. **Intelligent re-sizing**

Will re-size components to fill the browser and display on mobile devices.



3. **Support a wide variety of components**

htmlwidgets; base, lattice, and grid graphics; tabular data; gauges and value boxes; and text.



4. **Shiny compatible**

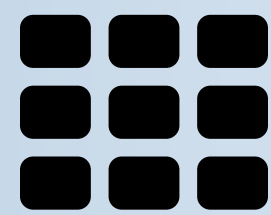


Outline



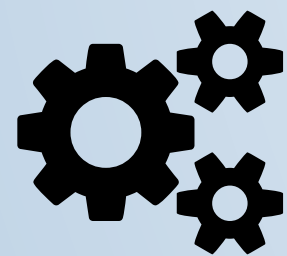
1. **Get Started**

How to install, open, and render a basic dashboard.



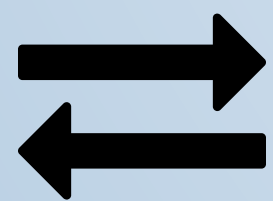
2. **Layouts**

Quick tour of columns, rows, pages, tabsets and storyboards



3. **Components**

Value boxes, gauges, and Shiny components

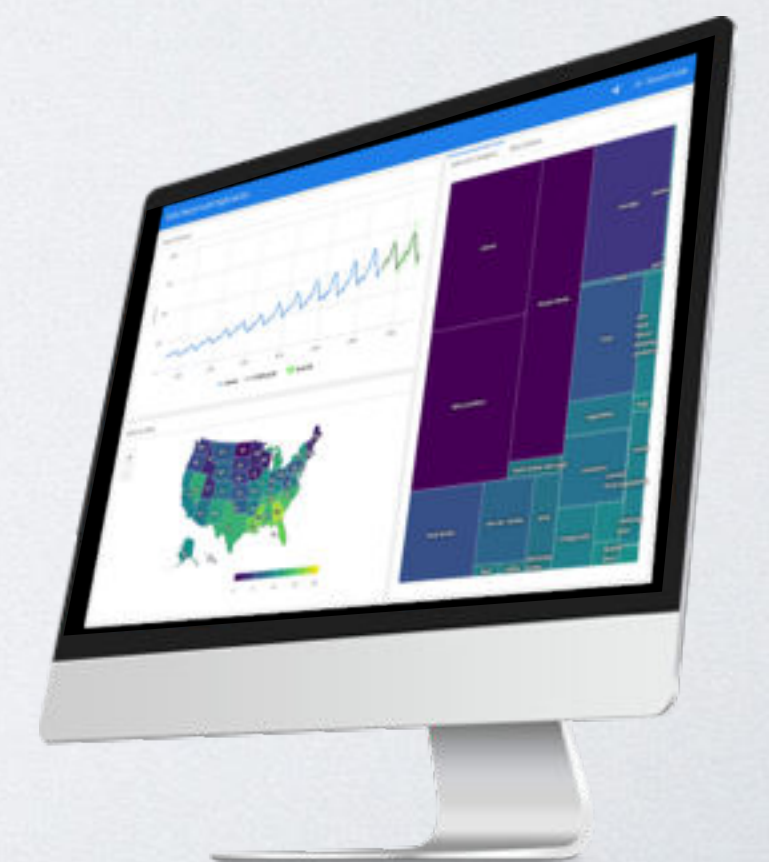


4. **flexdashboard vs. shinydashboard**

When should you use one or the other?



GET STARTED



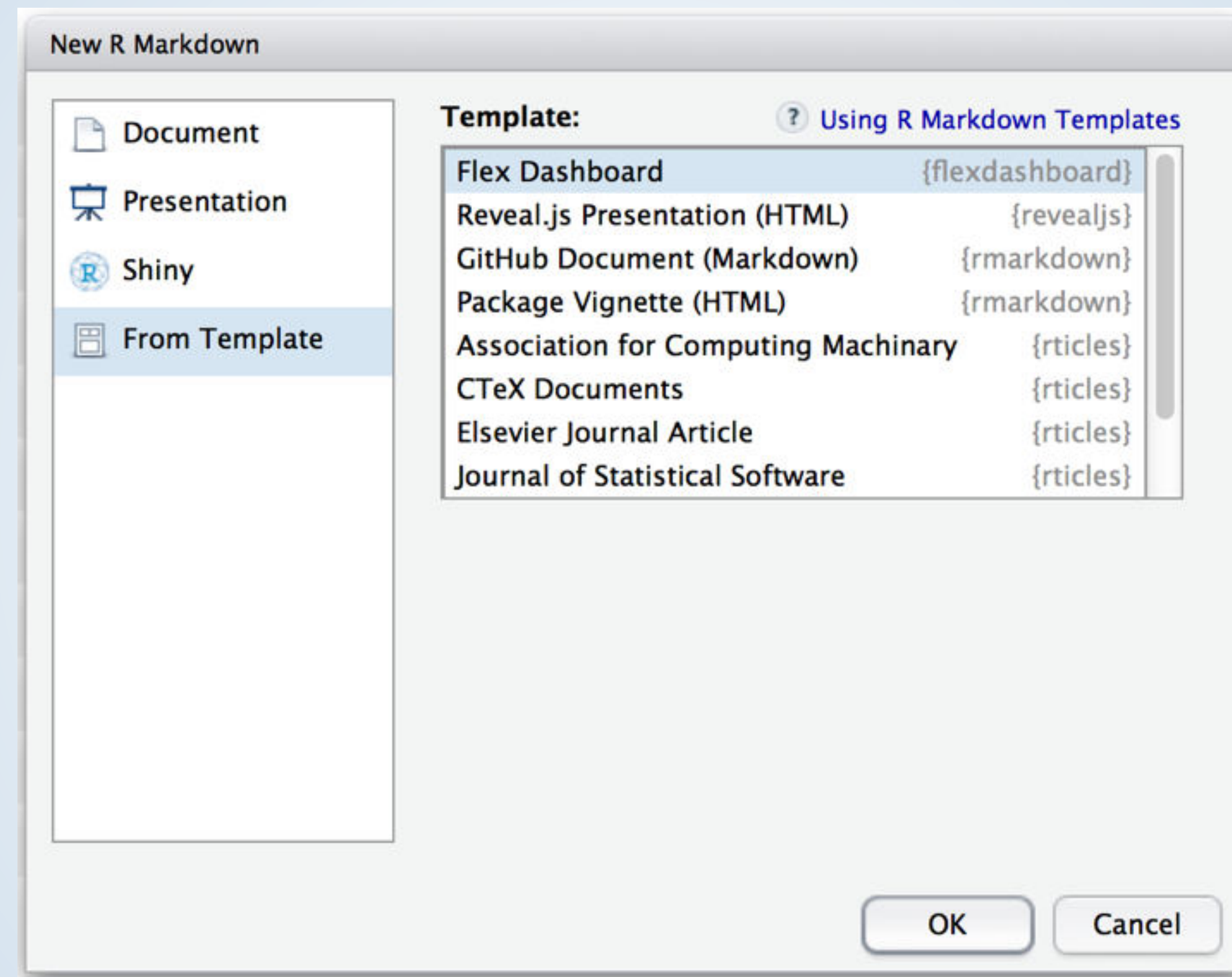
Install

Install the flexdashboard package from CRAN as follows:

```
install.packages("flexdashboard", type = "source")
```

Open new

File > New File > R Markdown... > From Template > Flex Dashboard



If you are not using RStudio, you can create a new flexdashboard R Markdown file from the R console:

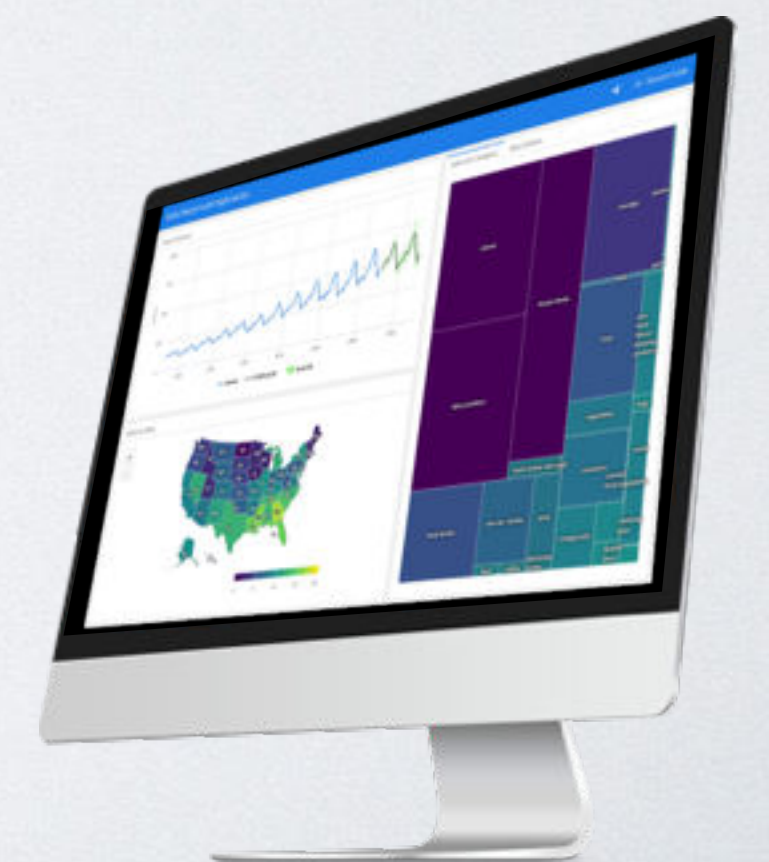
```
rmarkdown::draft("dashboard.Rmd", template = "flex_dashboard", package = "flexdashboard")
```

flexdashboard format

A flexdashboard is an .Rmd file with
output: flexdashboard::flex_dashboard

```
---  
title: "Untitled"  
output: flexdashboard::flex_dashboard  
---  
  
# Body
```


LAYOUTS



Layout Elements

| Header level | Alternative syntax | Layout element |
|--------------|--------------------|-------------------------------------|
| # | ===== | Page |
| ## | ----- | Column (or Row if orientation: row) |
| ### | | Box |

Modify elements with { } syntax, e.g.

```
## Column { .tabset .tabset-fade }
```

| | |
|--|--|
| <code>{.mobile}</code> | Indicates which of two identically named components will appear on a mobile device |
| <code>{.no-mobile}</code> | Component will not appear when displayed on a mobile device |
| <code>{.no-padding}</code> | Include no padding around charts (8 pixels included by default) |
| <code>{.no-title}</code> | Exclude the title of a component |
| <code>{.sidebar}</code> | Display component as a sidebar |
| <code>{.storyboard}</code> | Lays out page as a storyboard (also see <code>storyboard: true</code>) |
| <code>{.tabset}</code> | Display sub-components in a tabset |
| <code>{.tabset-fade}</code> | Adds fade in/out effect when switching tabs in a tabset |
| <code>{data-padding=10}</code> | Sets padding around charts (here to 10 pixels) |
| <code>{data-height=650}</code> | Sets relative height of a component |
| <code>{data-width=350}</code> | Sets relative width of a component |
| <code>{data-icon="fa-list"}</code> | Adds font awesome, Ionicons, or Bootstrap Glyphicons icon as page name in navbar |
| <code>{data-orientation=rows}</code> | Sets orientation for a page |
| <code>{data-navmenu="Menu A"}</code> | Indicates that a page should be accessed via a navbar menu (here named "Menu A") |
| <code>{data-commentary-width=400}</code> | Sets relative width of storyboard commentary component |

Output Options

Set output options for flex_dashboard as you would for other R Markdown formats,

New Line

Indent two
spaces

Indent four
spaces

```
---  
title: "Untitled"  
output:  
  flexdashboard::flex_dashboard:  
    orientation: rows  
    vertical_layout: scroll  
---  
  
# Body
```

Colon

YAML Output Options

| Option | Value(s) | Defines |
|------------------|--|--|
| css: | file path to .css | CSS to apply to dashboard |
| favicon: | file path to image | |
| logo: | file path to image | Logo to place in top left of navbar |
| navbar: | - {title: "A", href: "URL", align: left} | Links to manually add to navbar |
| orientation: | rows or columns | Orientation of columns within dashboard |
| social: | ["facebook", "twitter", "google-plus", "linkedin", "pinterest", and.or "menu"] | Links to social media to include in navbar |
| source_code: | embed or URL | Embeds or links to the source code |
| storyboard: | true or false | Display components as a storyboard |
| theme: | name of bootswatch theme | Bootswatch CSS theme to apply to dashboard |
| vertical_layout: | fill or scroll | Size elements to fit in window or not |

Storyboard Syntax

Each box in a storyboard becomes a frame with four parts:

One or two sentence
summary of the frame

Frame content

Optional line break

Optional commentary to
appear alongside frame in
sidebar to right

```
### Frame caption for navigation
```

```
```{r}
```

```
library(d3heatmap)
d3heatmap(mtcars, scale="column",
 colors="Blues")
```

```
```
```

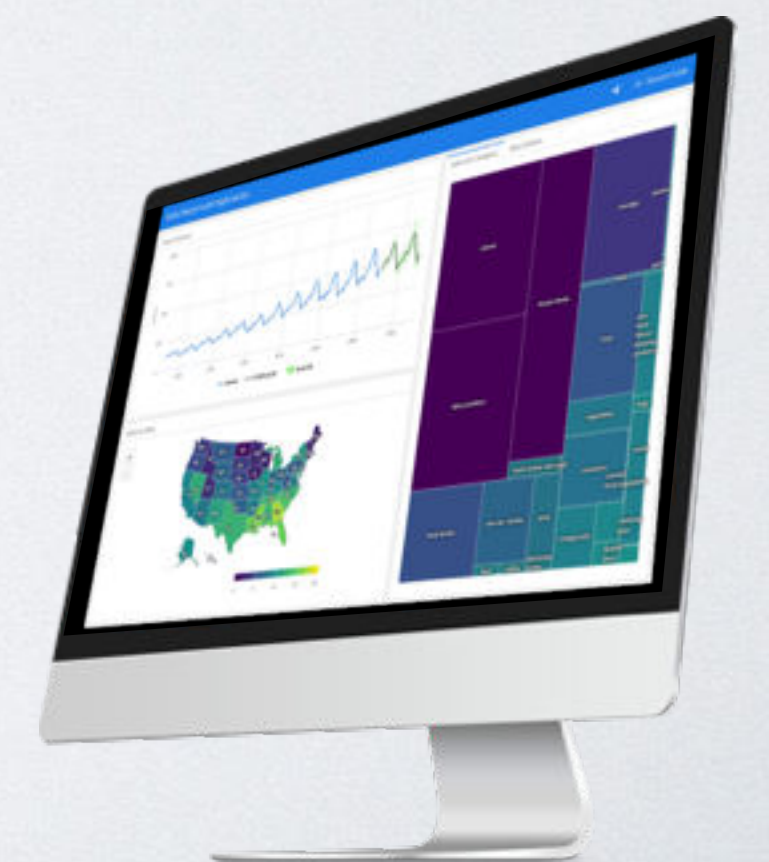
```
***
```

```
Some comment about this frame
```


storyboard: true

<https://beta.rstudioconnect.com/jjallaire/htmlwidgets-showcase-storyboard/>

COMPONENTS



valueBox()

6.2

Downloads per sec (last 5 min)



```
### Downloads per sec (last 5 min)
```

```
```{r}
```

```
valueBox(
```

```
 value = rate,
```

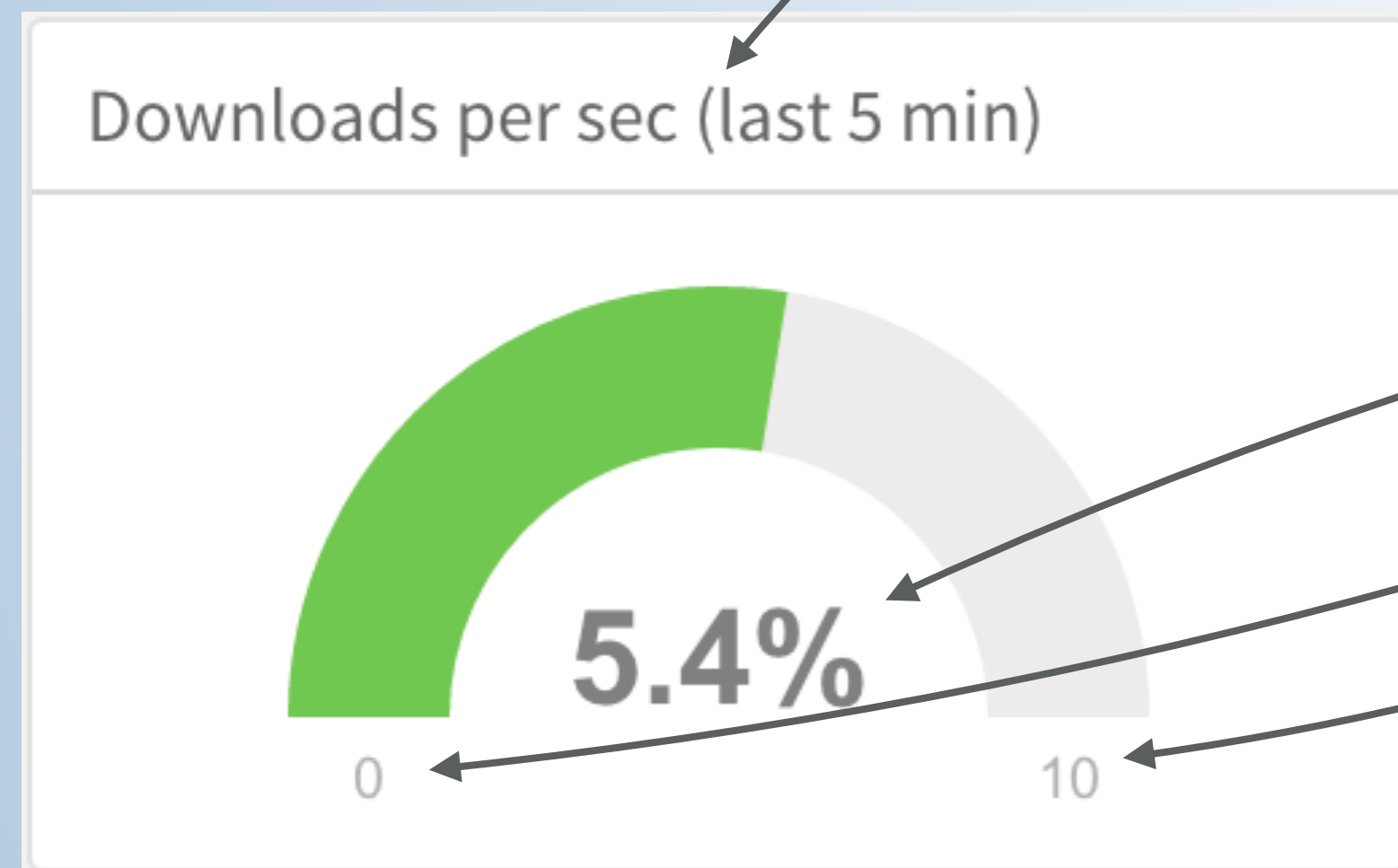
```
 icon = "fa-area-chart",
```

```
 color = if (rate >= 3) "warning" else "primary"
```

```
)
```

```
```
```

gauge()



```
### Downloads per sec (last 5 min)
```

```
```{r}  
gauge(
 value = rate,
 min = 0,
 max = 10,
 sectors = gaugeSectors(
 success = c(0, 6),
 warning = c(6, 8),
 danger = c(8, 10)
)
)
```
```


Shiny components

To add Shiny components to a flexdashboard, include `runtime: shiny` in the YAML

runtime: shiny

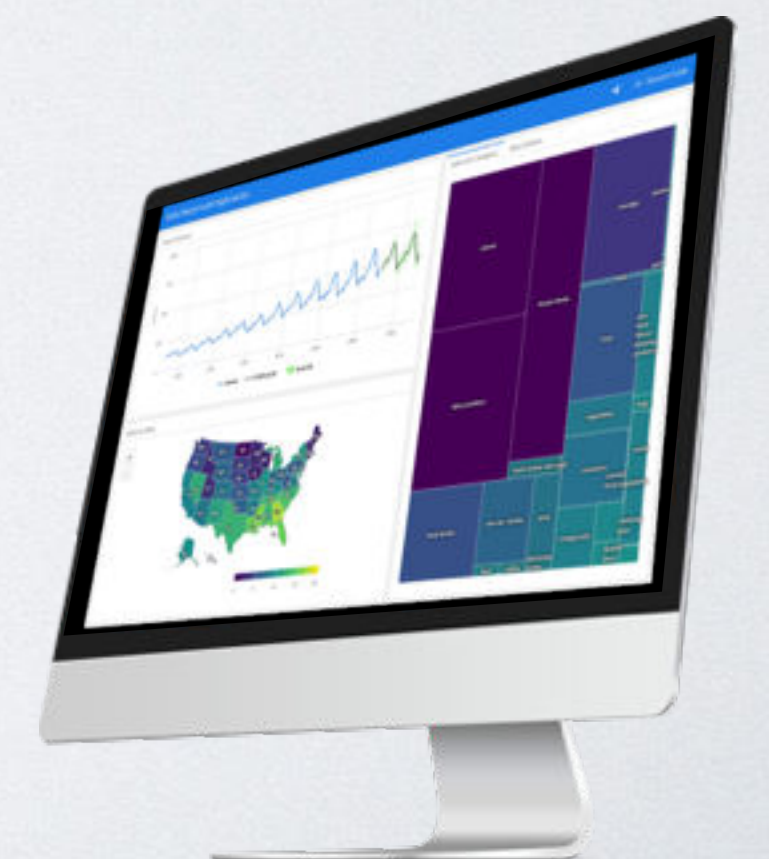
```
---  
title: "Untitled"  
output: flexdashboard::flex_dashboard  
runtime: shiny  
---
```

Render interactive valueBoxes and gauges with

```
renderValueBox()  
renderGauge()
```

<https://gallery.shinyapps.io/cran-gauge/>

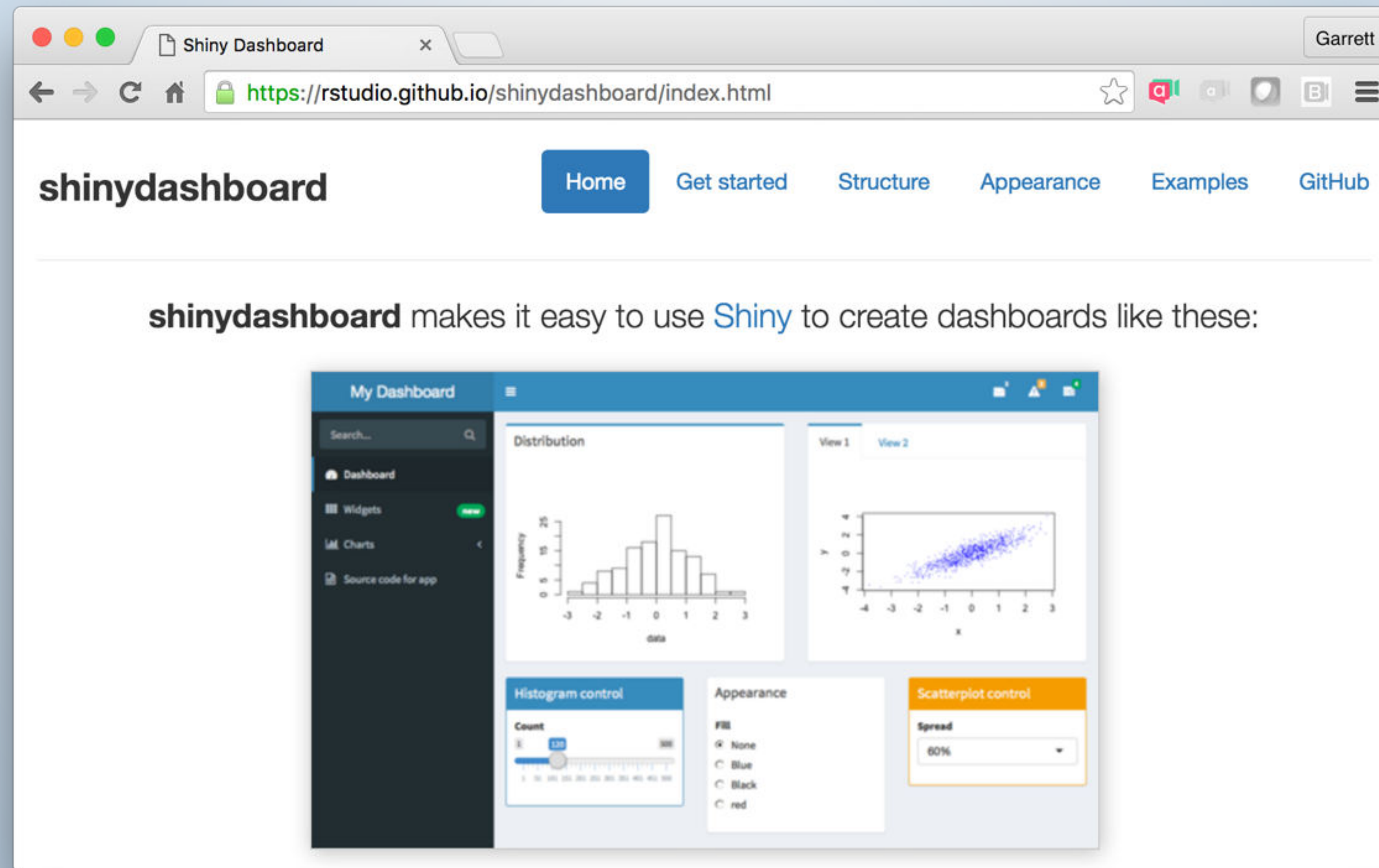
FLEXDASHBOARD VS. SHINYDASHBOARD



Shinydashboard

A dashboard package that works with Shiny's UI framework.

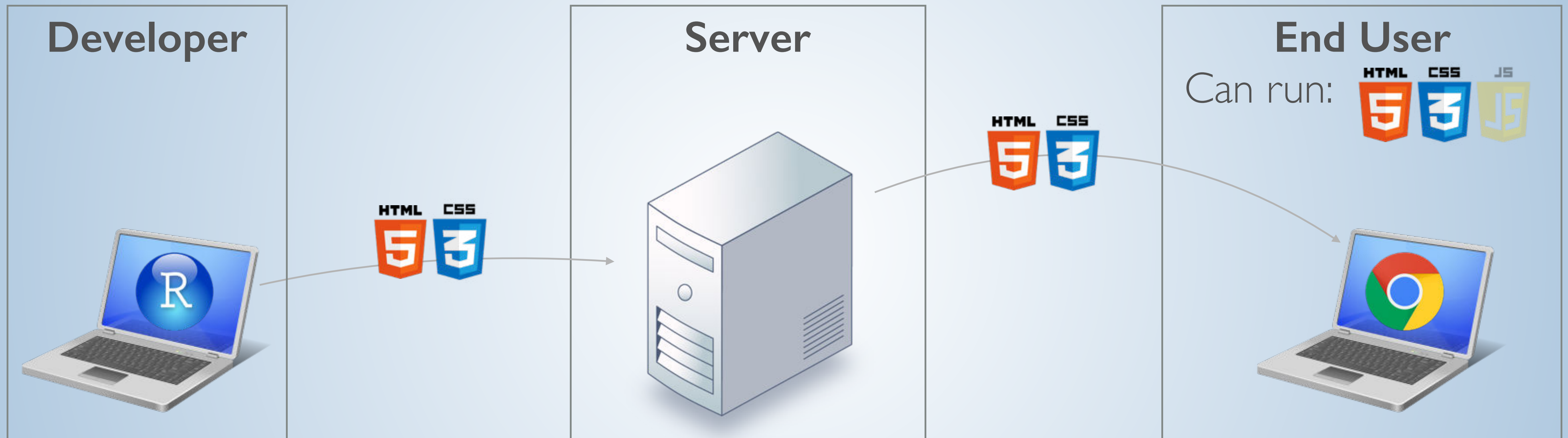
- * Directly integrated into Shiny (a Shiny dashboard is always a Shiny app)
- * More tools than flexdashboard, but more to learn



<https://gallery.shinyapps.io/087-crandash/>

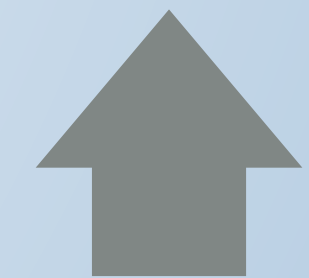
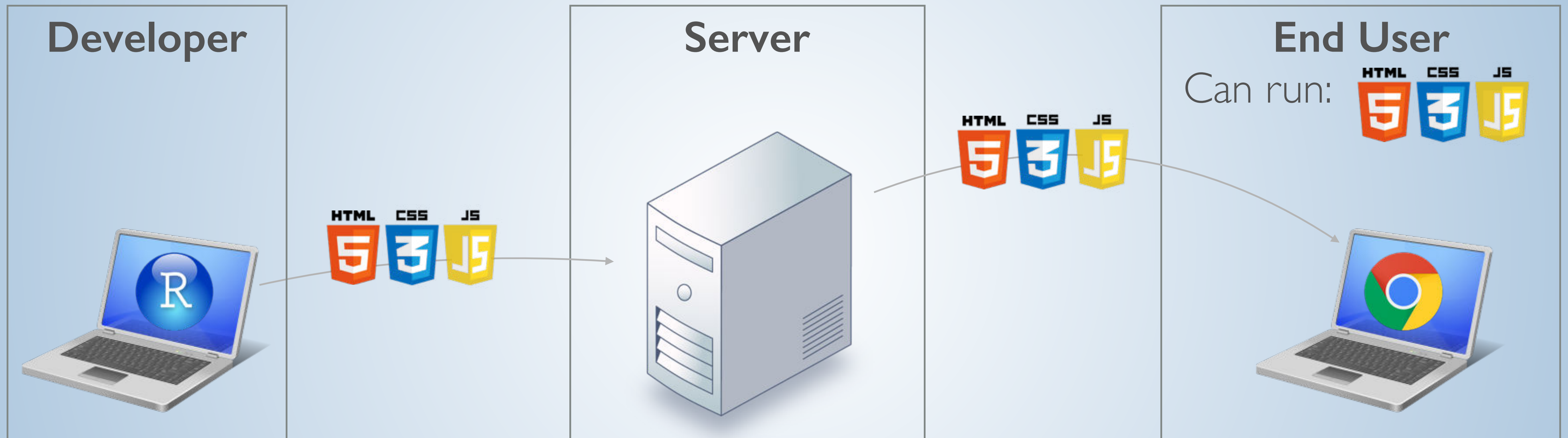
(Static) flexdashboard

A flexdashboard is a html DOM created with R Markdown



flexdashboard + htmlwidget

An htmlwidget is a self-contained JavaScript program



Processing happens in JavaScript
in user's web browser.
User needs the data.

flexdashboard + Shiny

A Shiny app is an html DOM and a set of instructions to run in R



Processing happens in R on the server.
Server needs Shiny Server (Pro).
User does not need the data.

flexdashboard vs. shinydashboard

| | Design experience | Data | Interactions | Server requirements |
|--|---------------------------|-------------------------------------|--------------------------------|--|
| Case 1 flexdashboard with htmlwidgets | Quick with constraints | Small and Public (easily shared) | Anything htmlwidgets can do | none |
| Case 2 flexdashboard with Shiny | Quick with constraints | Any size or Private | Anything R can do | Shiny Server or shinyapps.io |
| Case 3 shinydashboard | No constraints | Any size or Private | Anything R can do | Shiny Server or shinyapps.io |

Thank You

<https://rmarkdown.rstudio.com/flexdashboard/>

