BUILDING DASHBOARDS WITH FLEXDASHBOARD

Incorporating Shiny into Dashboards

Elaine McVey
Director of Quantitative Mobility
TransLoc

Why should I add Shiny? Or not?

Why

- **Interactivity**
- Lightweight

Why not

- Complication
- Hosting

Why should I add Shiny? Or not?

Why

- Interactivity
- **Lightweight**

Why not

- Complication
- Hosting

Why should I add Shiny? Or not?

Why

- Interactivity
- Lightweight

Why not

- **Complication**
- Hosting

Why should I add Shiny? Or not?

Why

- Interactivity
- Lightweight

Why not

- Complication
- **Hosting**

If not a Shiny app, then what?

A flexdashboard with Shiny is an *interactive RMarkdown document*

Making it shiny

runtime: shiny

BUILDING DASHBOARDS WITH FLEXDASHBOARD

Let's practice!

BUILDING DASHBOARDS WITH FLEXDASHBOARD

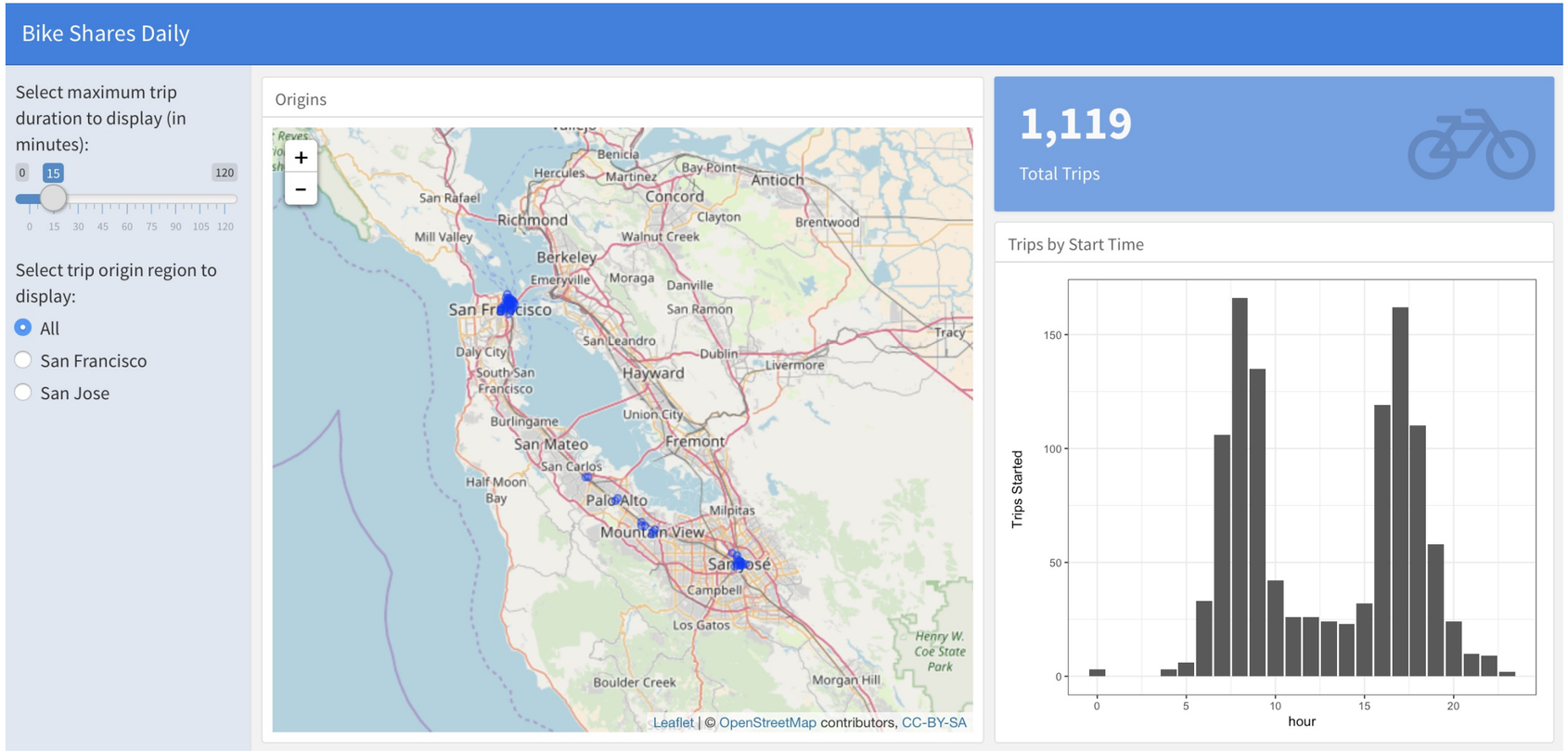
The Reactive Dataframe Pattern

Elaine McVey
Director of Quantitative Mobility
TransLoc

Creating a sidebar

Column {data-width=200 .sidebar}

Creating a sidebar



Adding user inputs

````
Column {data-width=200 .sidebar}
-----------------------------------------------------------------------

```{r}
sliderInput("duration_slider",
            label = "Select maximum trip duration to display (in minutes):",
            min = 0,
            max = 120,
            value = 15,
            step = 5,
            dragRange = TRUE)
```
````

Making our dataframe reactive

```
```{r}
sliderInput("duration_slider",
            label = "Select maximum trip duration to display (in minutes):",
            min = 0,
            max = 120,
            value = 15,
            step = 5,
            dragRange = TRUE)

show_trips_df <- reactive({

  trips_df %>%
    filter(duration_sec <= input$duration_slider * 60)

})
```
```

Using the reactive dataframe

````
Column {data-width=450}
-----------------------------------------------------------------------

### Origins

```{r}

  trips_df %>%
    rename(latitude = start_latitude,
           longitude = start_longitude) %>%
    group_by(start_station_id, latitude, longitude) %>%
    count() %>%
    leaflet() %>%
    addTiles() %>%
    addCircles(radius = ~n)

```
````

````
Column {data-width=450}
-----------------------------------------------------------------------

### Origins

```{r}

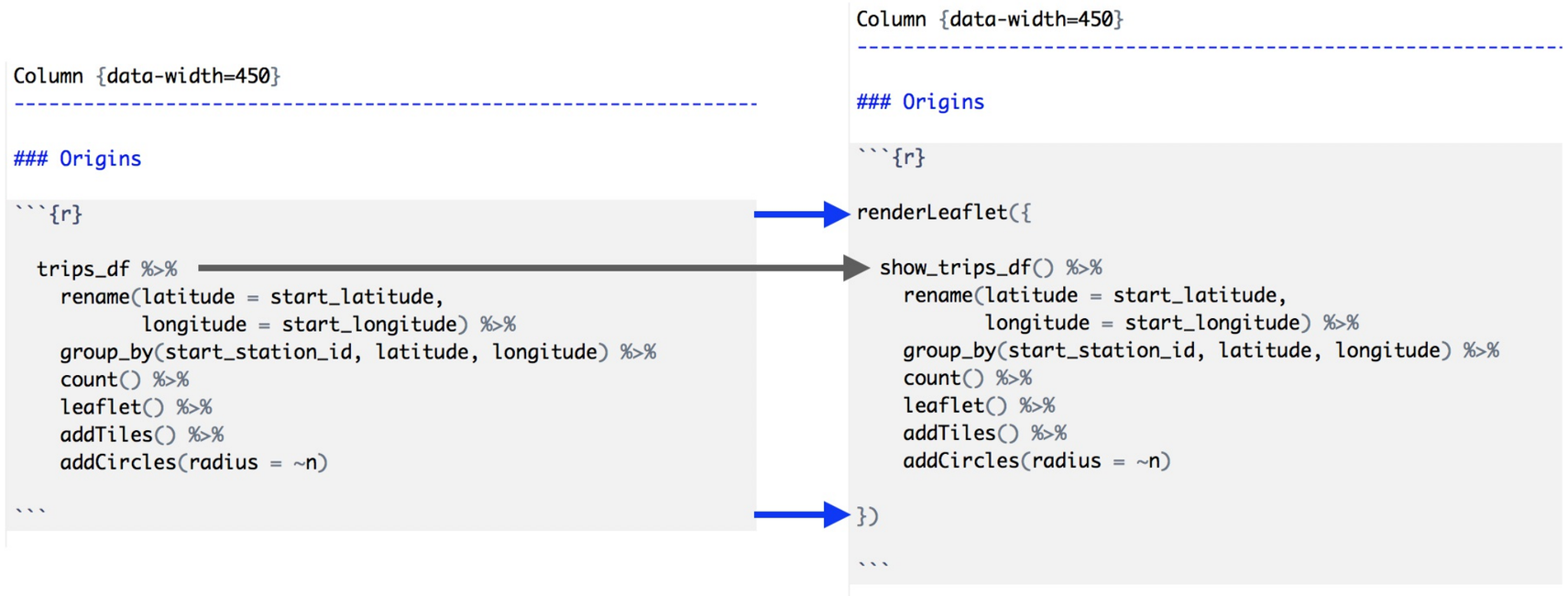
renderLeaflet({

  show_trips_df() %>%
    rename(latitude = start_latitude,
           longitude = start_longitude) %>%
    group_by(start_station_id, latitude, longitude) %>%
    count() %>%
    leaflet() %>%
    addTiles() %>%
    addCircles(radius = ~n)

})

```
````

Making dashboard components reactive



```
Column {data-width=450}
-----------------------------------------------------------------------

### Origins

```{r}

  trips_df %>%
    rename(latitude = start_latitude,
           longitude = start_longitude) %>%
    group_by(start_station_id, latitude, longitude) %>%
    count() %>%
    leaflet() %>%
    addTiles() %>%
    addCircles(radius = ~n)

```
```

```
Column {data-width=450}
-----------------------------------------------------------------------

### Origins

```{r}

renderLeaflet({

  show_trips_df() %>%
    rename(latitude = start_latitude,
           longitude = start_longitude) %>%
    group_by(start_station_id, latitude, longitude) %>%
    count() %>%
    leaflet() %>%
    addTiles() %>%
    addCircles(radius = ~n)

})

```
```

Steps to the reactive dataframe pattern

1. Create a sidebar column (using .sidebar).
2. Add user inputs to the sidebar (using xyzInput() Shiny widgets).
3. Make a "dataframe" that reacts to user inputs (using reactive()).
4. Replace the dataframe in the dashboard component code with the reactive version.
5. Wrap each dashboard output with the appropriate Shiny version (renderXyz()).

BUILDING DASHBOARDS WITH FLEXDASHBOARD

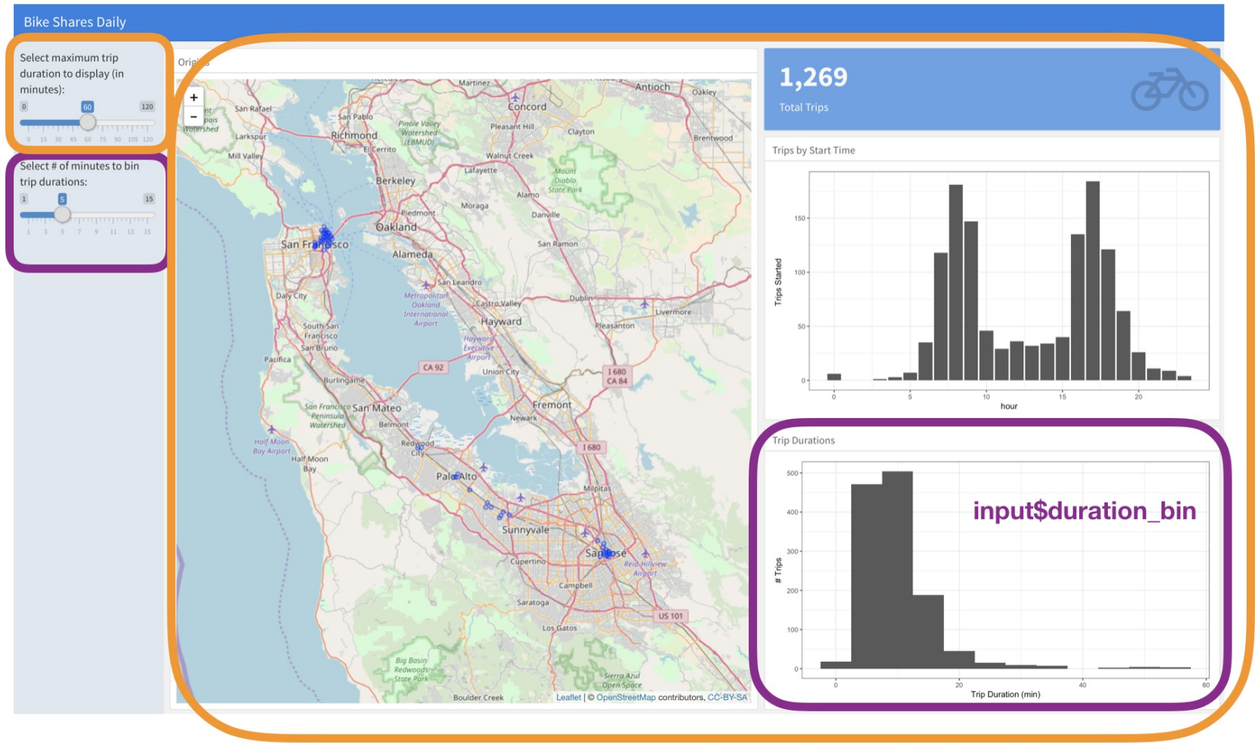
Let's practice!

BUILDING DASHBOARDS WITH FLEXDASHBOARD

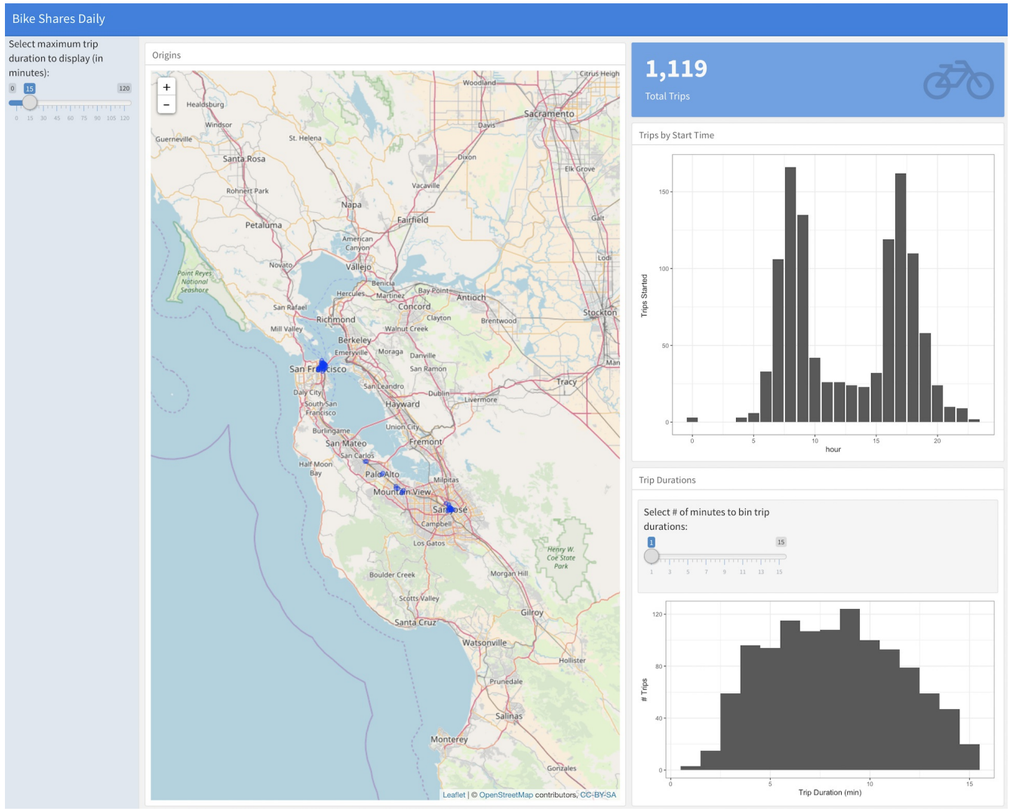
Customized Inputs for Charts

Elaine McVey
Director of Quantitative Mobility
TransLoc

Chart-Specific Effects



Moving Inputs Into Charts



Moving Inputs Into Charts

```
```{r}

fillCol(height = 600, flex = c(NA, 1),
  inputPanel(
    sliderInput("xyz_input", ...)
  ),
  plotOutput("xyzPlot", height = "100%")
)

output$xyzPlot <- renderPlot({

})

```
```

Moving Inputs Into Charts

```
```{r}

fillCol(height = 600, flex = c(NA, 1),
  inputPanel(
    sliderInput("xyz_input", ...)
  ),
  plotOutput("xyzPlot", height = "100%")
)

output$xyzPlot <- renderPlot({

})

```
```

Moving Inputs Into Charts

```
```{r}

fillCol(height = 600, flex = c(NA, 1),
  inputPanel(
    sliderInput("xyz_input", ...)
  ),
  plotOutput("xyzPlot", height = "100%")
)

output$xyzPlot <- renderPlot({

})

```
```

Moving Inputs Into Charts

```
```{r}

fillCol(height = 600, flex = c(NA, 1),
  inputPanel(
    sliderInput("xyz_input", ...)
  ),
  plotOutput("xyzPlot", height = "100%")
)

output$xyzPlot <- renderPlot({

})

```
```

A Shortcut

```
Global Sidebar {.sidebar}
=====================================

```{r}

```

Overview
=====================================

Column {data-width=650 .tabset}

Origins
```

BUILDING DASHBOARDS WITH FLEXDASHBOARD

Let's practice!

BUILDING DASHBOARDS WITH FLEXDASHBOARD

Course Recap

Elaine McVey
Director of Quantitative Mobility
TransLoc

Resources

- https://rmarkdown.rstudio.com/flexdashboard/
- https://www.htmlwidgets.org/

Resources

- **https://rmarkdown.rstudio.com/flexdashboard/**
- https://www.htmlwidgets.org/

Resources

- https://rmarkdown.rstudio.com/flexdashboard/
- **https://www.htmlwidgets.org/**
  - leaflet
  - DT (datatable)
  - plotly
  - highcharter

shinydashboard

BUILDING DASHBOARDS WITH FLEXDASHBOARD

Thank you!