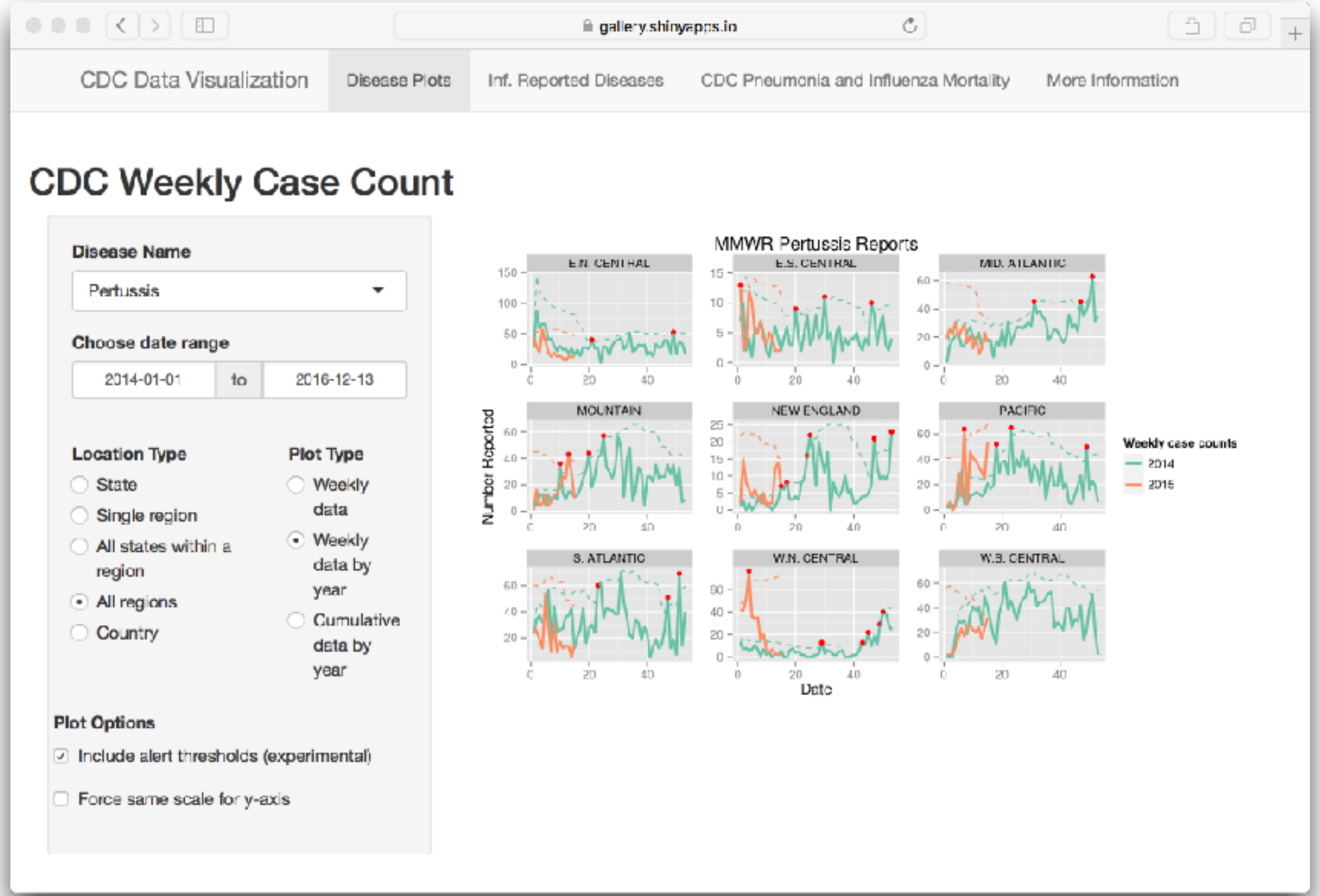
Introducing bookmarkable state for Shiny

Winston Chang

2016–12–14

RStudio



CDC Weekly Case Count

Disease Name

Pertussis

Choose date range

2014-01-01 to 2016-12-13

Location Type

○ State
○ Single region
○ All states within a region
● All regions
○ Country

Plot Type

○ Weekly data
● Weekly data by year
○ Cumulative data by year

Plot Options

☑ Include alert thresholds (experimental)

□ Force same scale for y-axis

https://gallery.shinyapps.io/CDCPlot/

To: Boss <boss@megacorp.com>

Subject: Pertussis data

Hi Boss, here's the app that shows the latest data. To see the results:

Go to https://gallery.shinyapps.io/CDCPlot/

For "Disease Name", select "Pertussis".

For "Choose date range", make it start from 2014-01-01.

For "Plot Type", choose "Weekly data by year".

For "Plot Options" check "Include alert thresholds".

Then you should see what I'm seeing.

Enjoy your holidays!

To: Boss <boss@megacorp.com>

Subject: Pertussis data

Hi Boss, here's the latest data:

https://gallery.shinyapps.io/CDCPlot/?_state_id_=129a0b4c

Enjoy your holidays!

Bookmarkable state

(Requires Shiny 0.14)

app.R — A very simple app

```r
ui <- fluidPage(
  textInput("txt", "Enter text"),
  verbatimTextOutput("out")
)
server <- function(input, output, session) {
  output$out <- renderText({
    input$txt
  })
}

shinyApp(ui, server)
```

app.R — With bookmarking

```r
ui <- function(request) {
  fluidPage(
    textInput("txt", "Enter text"),
    verbatimTextOutput("out"),
    bookmarkButton()
  )
}
server <- function(input, output, session) {
  output$out <- renderText({
    input$txt
  })
}

shinyApp(ui, server, enableBookmarking = "url")
```

ui.R + server.R + global.R

```
## ui.R ##
function(request) {
  fluidPage(
    textInput("txt", "Enter text"),
    verbatimTextOutput("out"),
    bookmarkButton()
  )
}

## server.R ##
function(input, output, session) {
  output$out <- renderText({
    input$txt
  })
}

## global.R ##
enableBookmarking("url")
```

Live deployed example:

https://gallery.shinyapps.io/113-bookmarking-url/

URL-encoded bookmarks

https://domain.com/app/**?_inputs_&n=123&txt="abc"**

app.R: saved-to-server bookmarks

```r
ui <- function(request) {
  fluidPage(
    textInput("txt", "Enter text"),
    verbatimTextOutput("out"),
    bookmarkButton()
  )
}
server <- function(input, output, session) {
  output$out <- renderText({
    input$txt
  })
}

shinyApp(ui, server, enableBookmarking = "server")
```

URL-encoded	Saved-to-server
No state on server	State saved on server
URLs may be long	Always short URL
URL length limits data (~2000 chars in some browsers)	No limits to data
Can't store uploaded files	Can store uploaded files
Any hosting platform	Requires Shiny Server (for now)

Deploying bookmarkable apps

Hosting platform	URL-encoded	Saved-to-server
Local computer (development)	✓	✓
Shiny Server Pro	✓	✓
Shiny Server Open Source	✓	✓
shinyapps.io	✓	In progress
RStudio Connect	✓	In progress

Advanced bookmarking

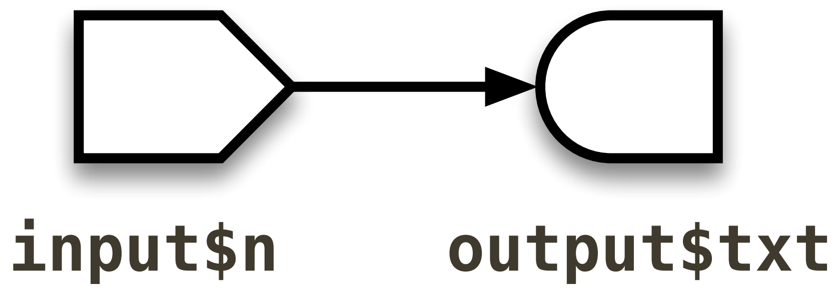
Fully input-dependent applications:

The state of the outputs at time **t** is fully determined by the state of the inputs at time **t**.

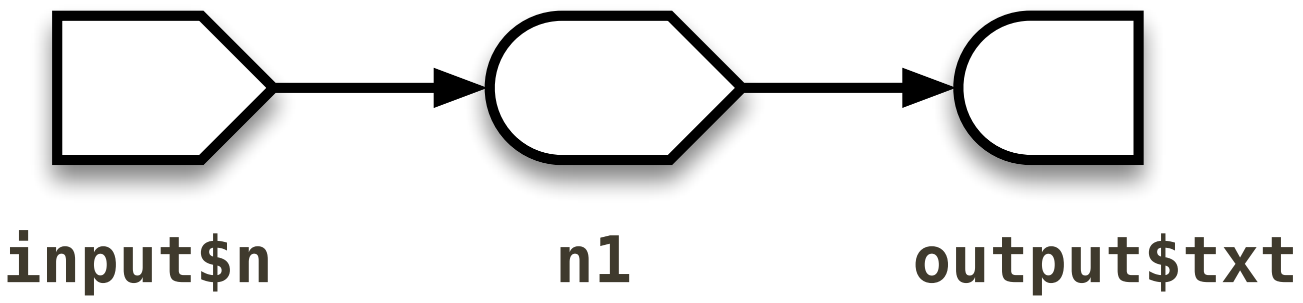
For these applications, bookmarking should Just Work.

Fully input-dependent applications

```
output$txt <- renderText(input$n)
```



```
n1 <- reactive(input$n + 1)

output$txt <- renderText(n1())
```



Partly input-dependent applications:

The state of the outputs at time **t** is only **partly** determined by the state of the inputs at time **t**. Other things may influence it, including inputs at previous times, or external data.

These applications will require a bit more code for bookmarking.

Partly input-dependent application

```
ui <- function(request) {
  fluidPage(
    sidebarPanel(
      sliderInput("n", "Value to add", min = 0, max = 100, value = 50),
      actionButton("add", "Add"), br(), br(),
      bookmarkButton()
    ),
    mainPanel(
      h3("Sum:", textOutput("sum"))
    )
  )
}
server <- function(input, output, session) {
  vals <- reactiveValues(sum = 0)

  observeEvent(input$add, {
    vals$sum <- vals$sum + input$n
  })

  output$sum <- renderText({
    vals$sum
  })
}

shinyApp(ui, server, enableBookmarking = "url")
```

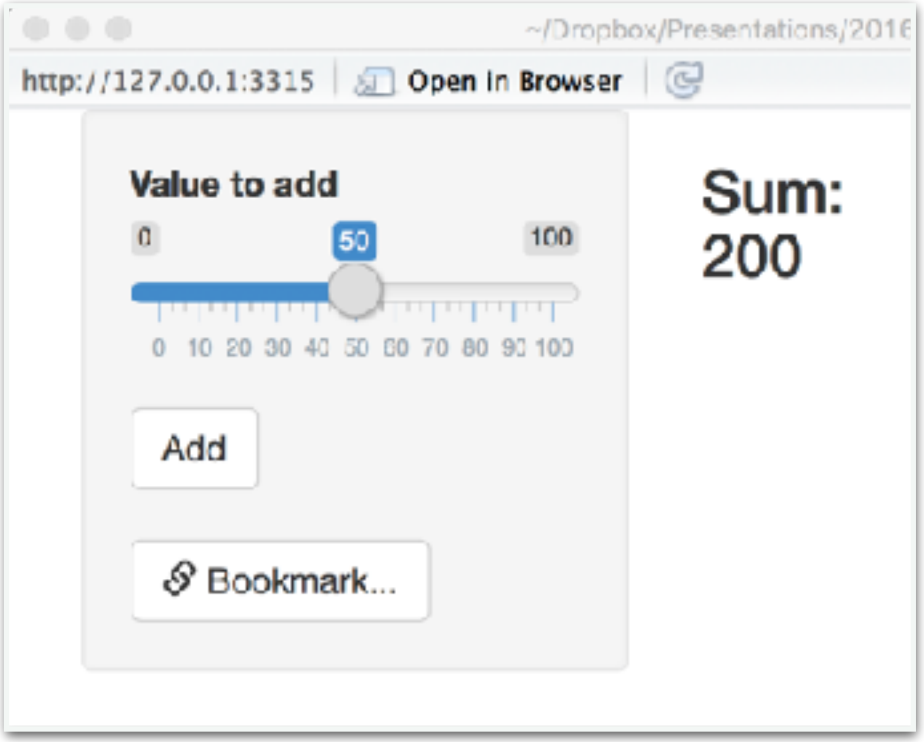
A partly input-dependent application

```
server <- function(input, output, session) {
  vals <- reactiveValues(sum = 0)

  observeEvent(input$add, {
    vals$sum <- vals$sum + input$n
  })
  output$sum <- renderText({
    vals$sum
  })
}
```

Each time the "Add" button is clicked…

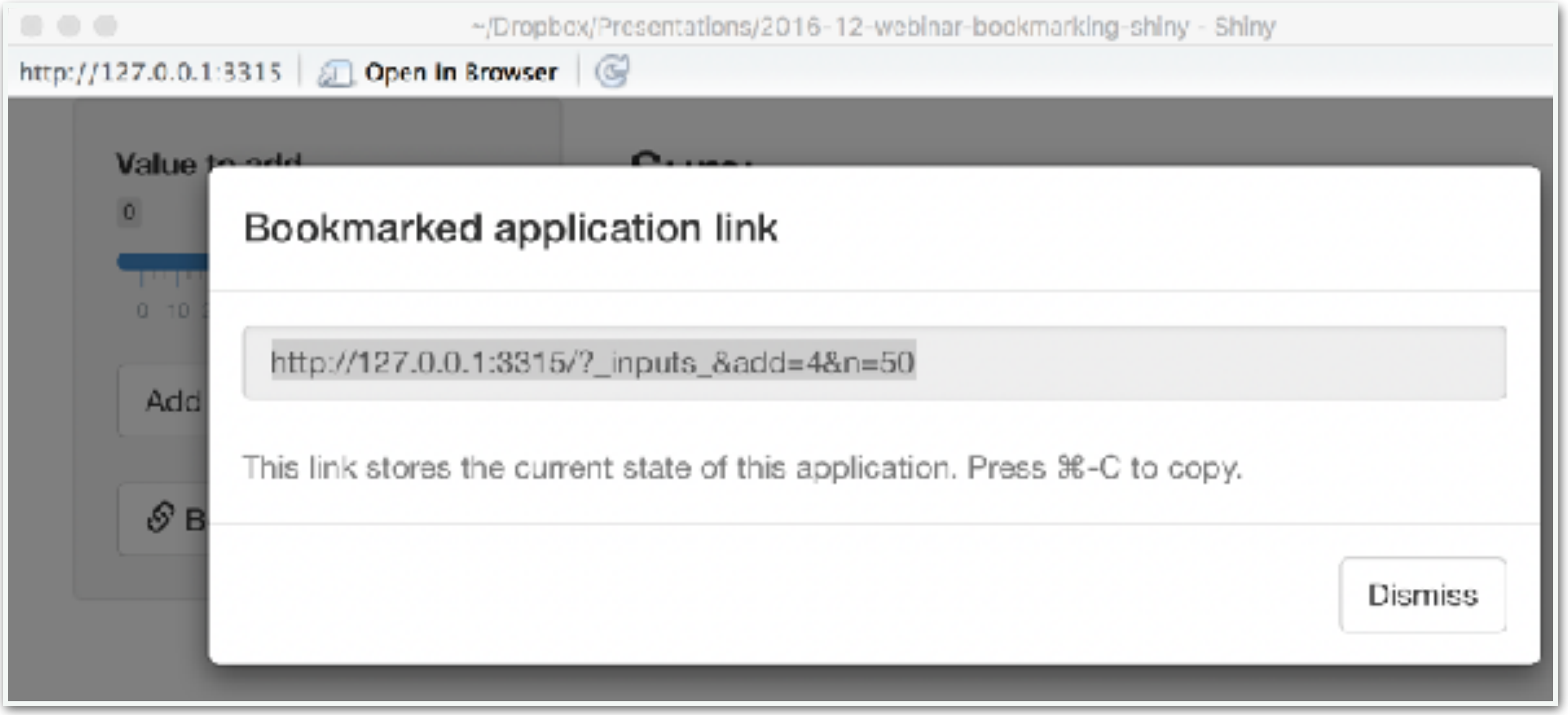
…add input$n to vals$sum…

…then this renderText re-executes



Sum:
200

Bookmark

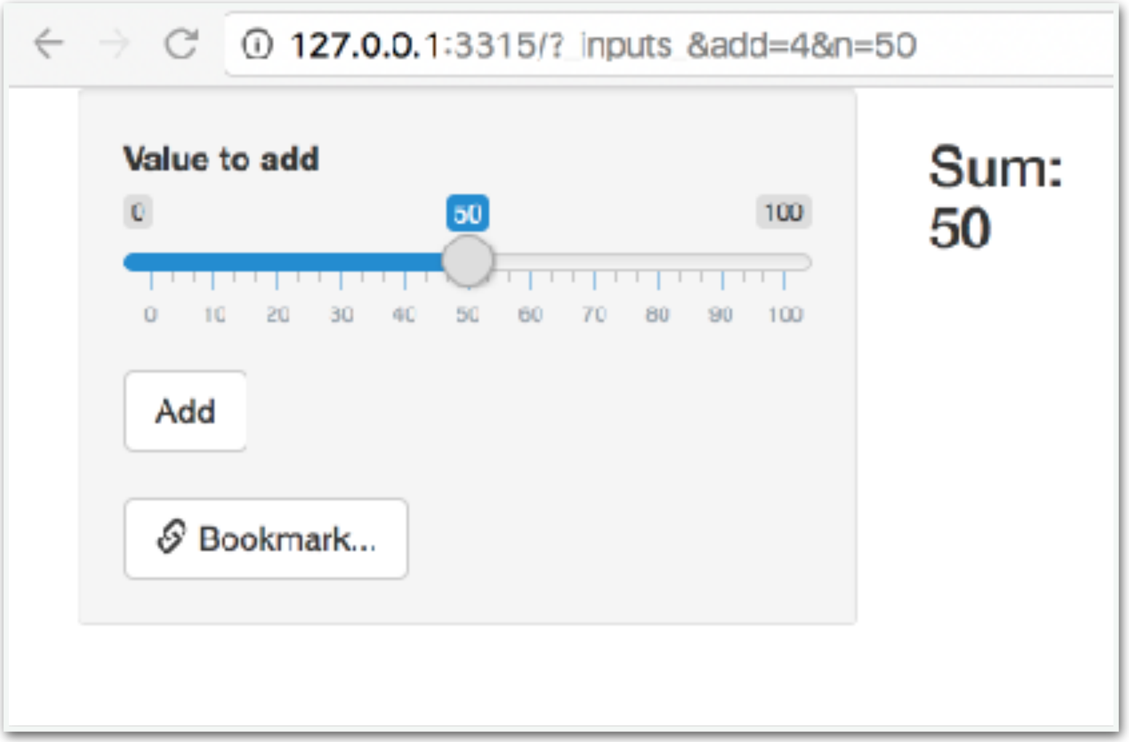

~/Dropbox/Presentations/2016-12-webinar-bookmarking-shiny - Shiny
http://127.0.0.1:3315
Open in Browser
Value to add
0
0 10
Add
B
Sum:

Bookmarked application link

http://127.0.0.1:3315/?_inputs_&add=4&n=50

This link stores the current state of this application. Press ⌘-C to copy.

Restore



Sum:
50

Bookmark/Restore callbacks

```
server <- function(input, output, s
  vals <- reactiveValues(sum = 0)

  onBookmark(function(state) {
    state$values$currentSum <- vals$sum
  })
  onRestore(function(state) {
    vals$sum <- state$values$currentSum
  })

  observeEvent(input$add, {
    vals$sum <- vals$sum + input$n
  })
  output$sum <- renderText({
    vals$sum
  })
}
```

When app is bookmarked, save vals$sum in the bookmark state values

When app is restored, retrieve vals$sum from the bookmark state values

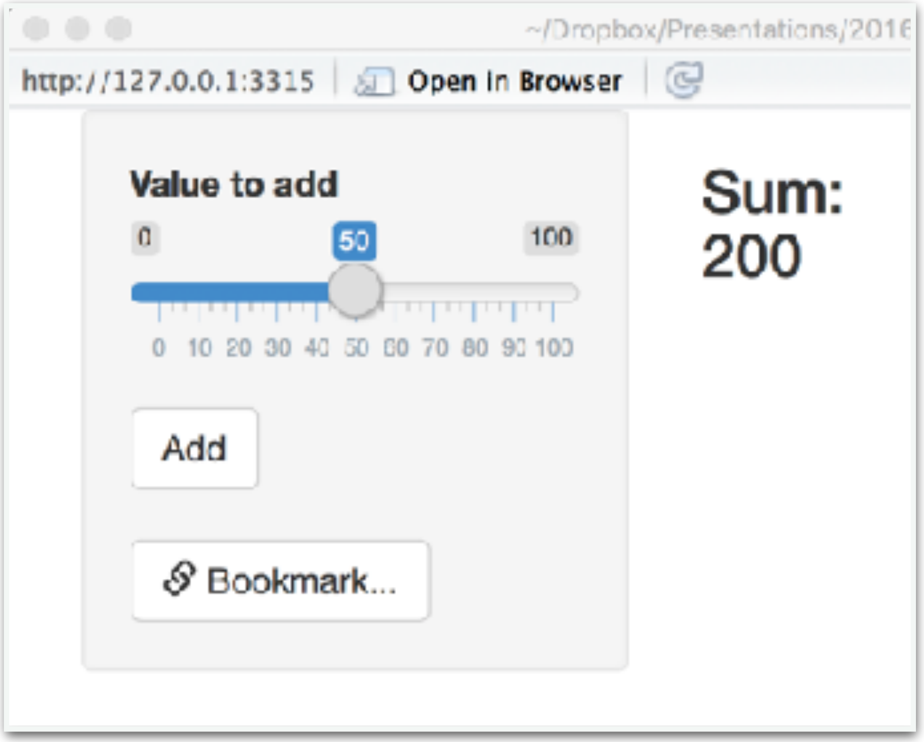
Bookmarking callbacks

```
server <- function(input, output, session) {
  vals <- reactiveValues(sum = 0)

  onBookmark(function(state) {
    state$values$currentSum <- vals$sum
  })
  onRestore(function(state) {
    vals$sum <- state$values$curr
  })

  observeEvent(input$add, {
    vals$sum <- vals$sum + input$n
  })
  output$sum <- renderText({
    vals$sum
  })
}
```

After app is restored, input$add has a value (like 4), so it triggers this observer

One solution:

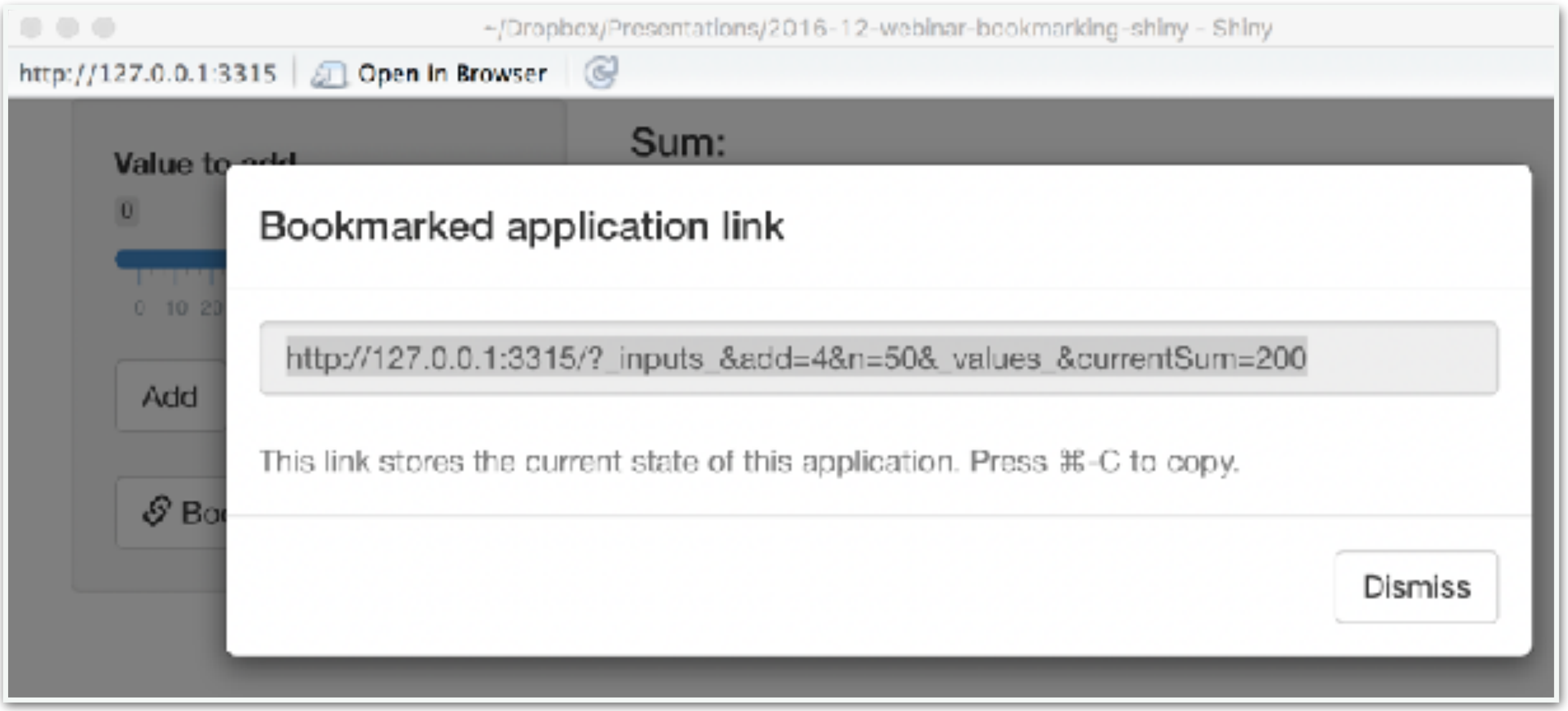
Don't bookmark the input$add button



http://127.0.0.1:3315

Sum:
200

Bookmark


~/Dropbox/Presentations/2016-12-webinar-bookmarking-shiny - Shiny
Open in Browser
Value to add
0
0 10 20
Add
Bookmark...
Sum:
Dismiss

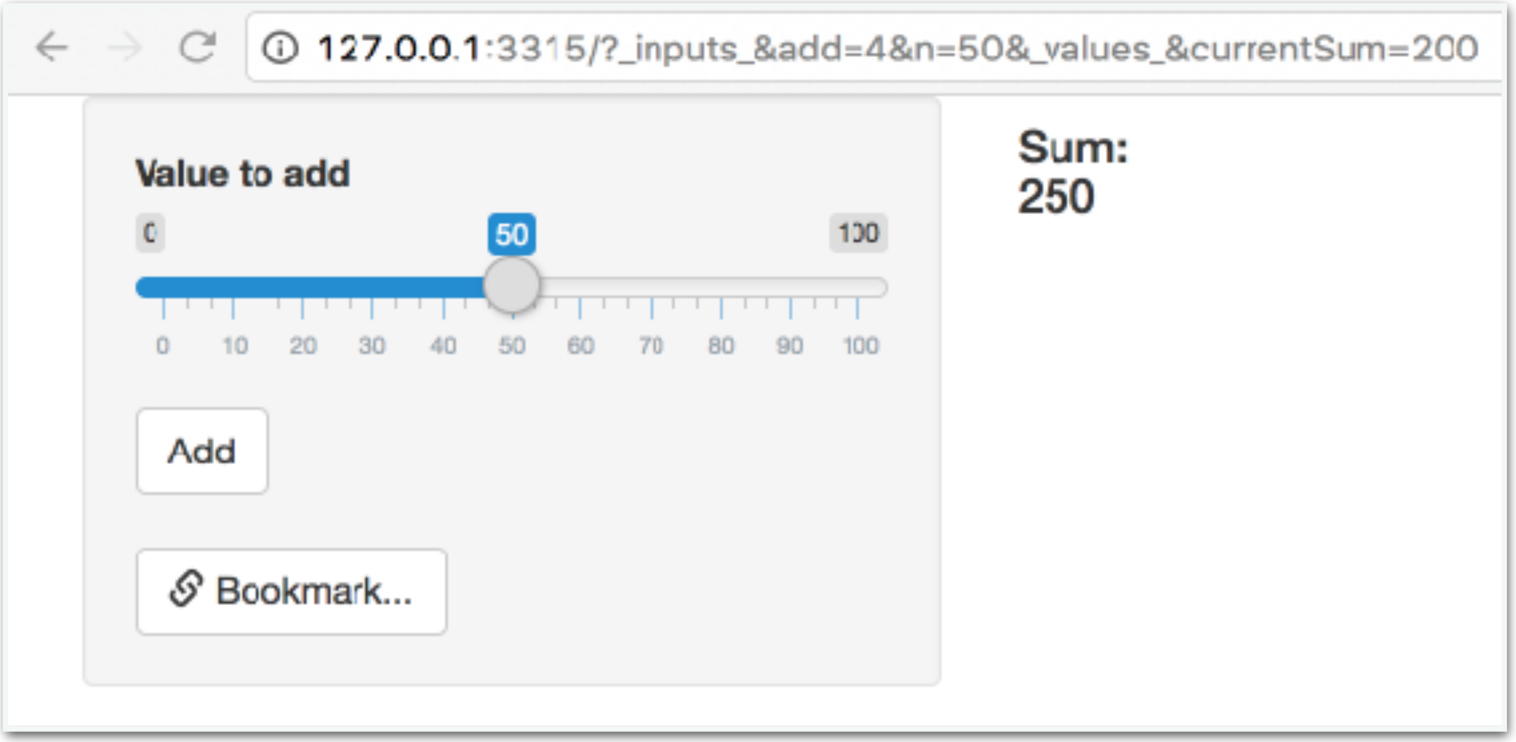
http://127.0.0.1:3315

Bookmarked application link

http://127.0.0.1:3315/?_inputs_&add=4&n=50&_values_&currentSum=200

This link stores the current state of this application. Press ⌘-C to copy.

Restore



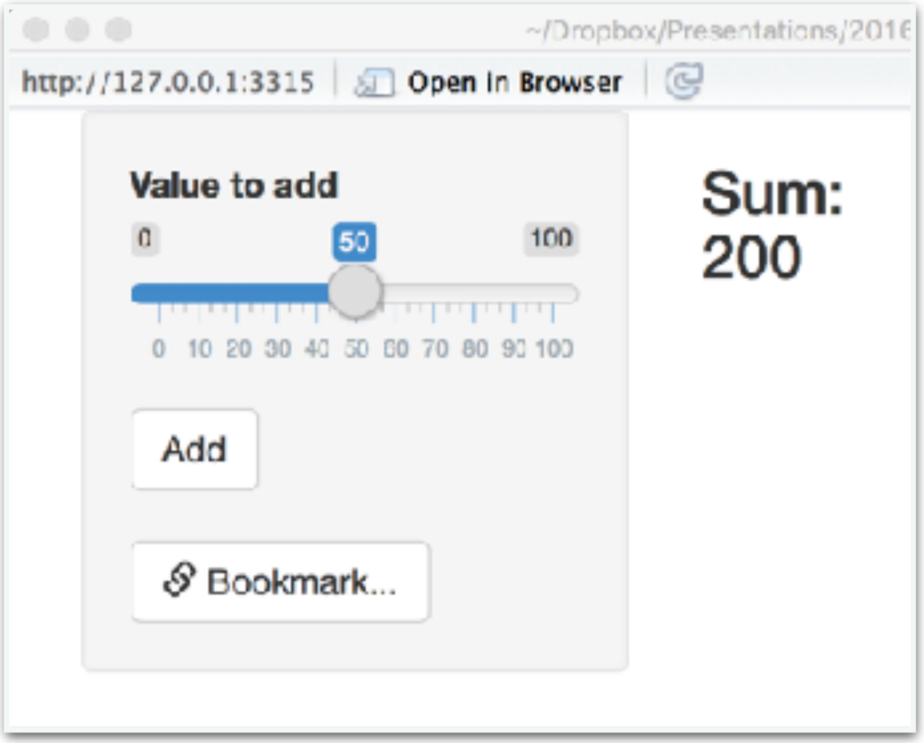
127.0.0.1:3315/?_inputs_&add=4&n=50&_values_&currentSum=200

Sum:
250

Excluding inputs from bookmarking

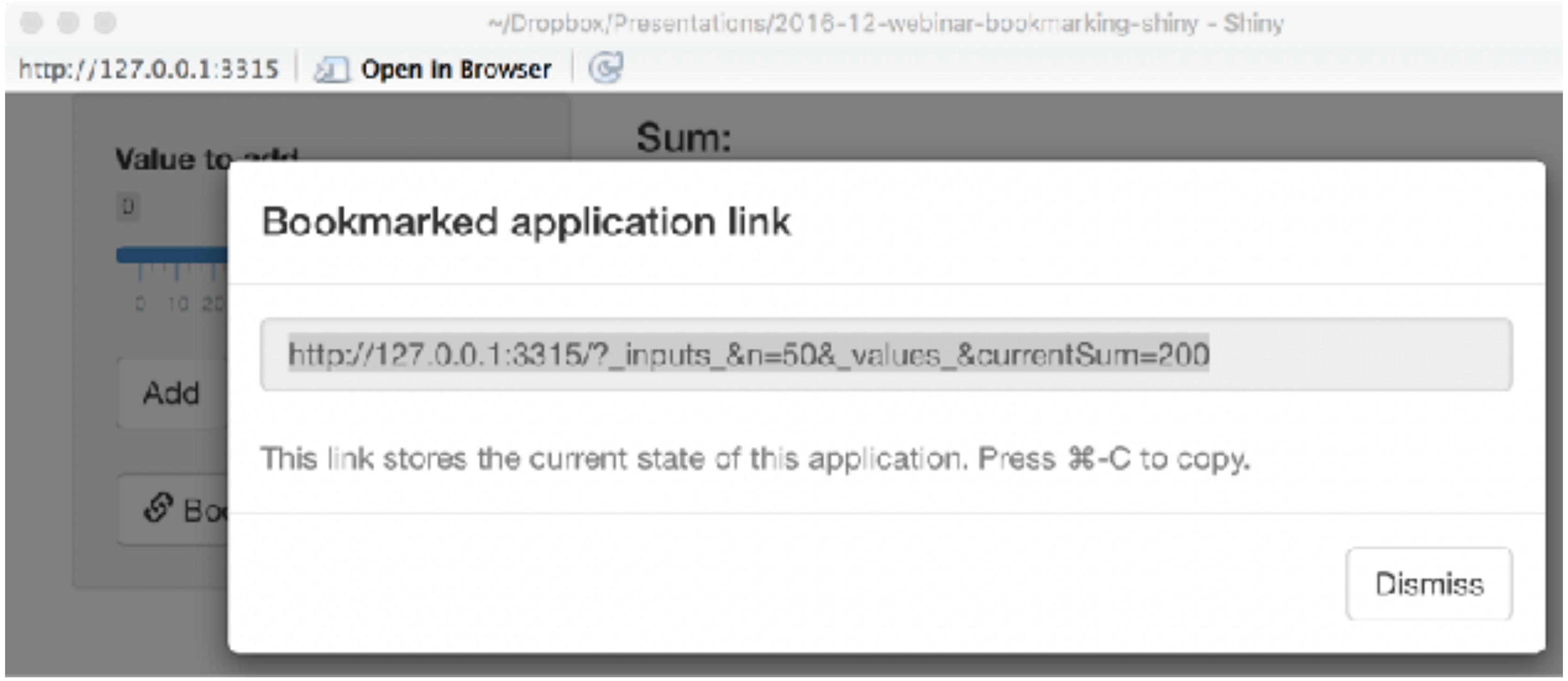
```
server <- function(input, output, session) {
  vals <- reactiveValues(sum = 0)

  onBookmark(function(state) {
    state$values$currentSum <- vals$sum
  })
  onRestore(function(state) {
    vals$sum <- state$values$currentSum
  })
  setBookmarkExclude("add")

  observeEvent(input$add, {
    vals$sum <- vals$sum + input$n
  })
  output$sum <- renderText({
    vals$sum
  })
}
```

Don't bookmark input$add



Sum:
200

Bookmark

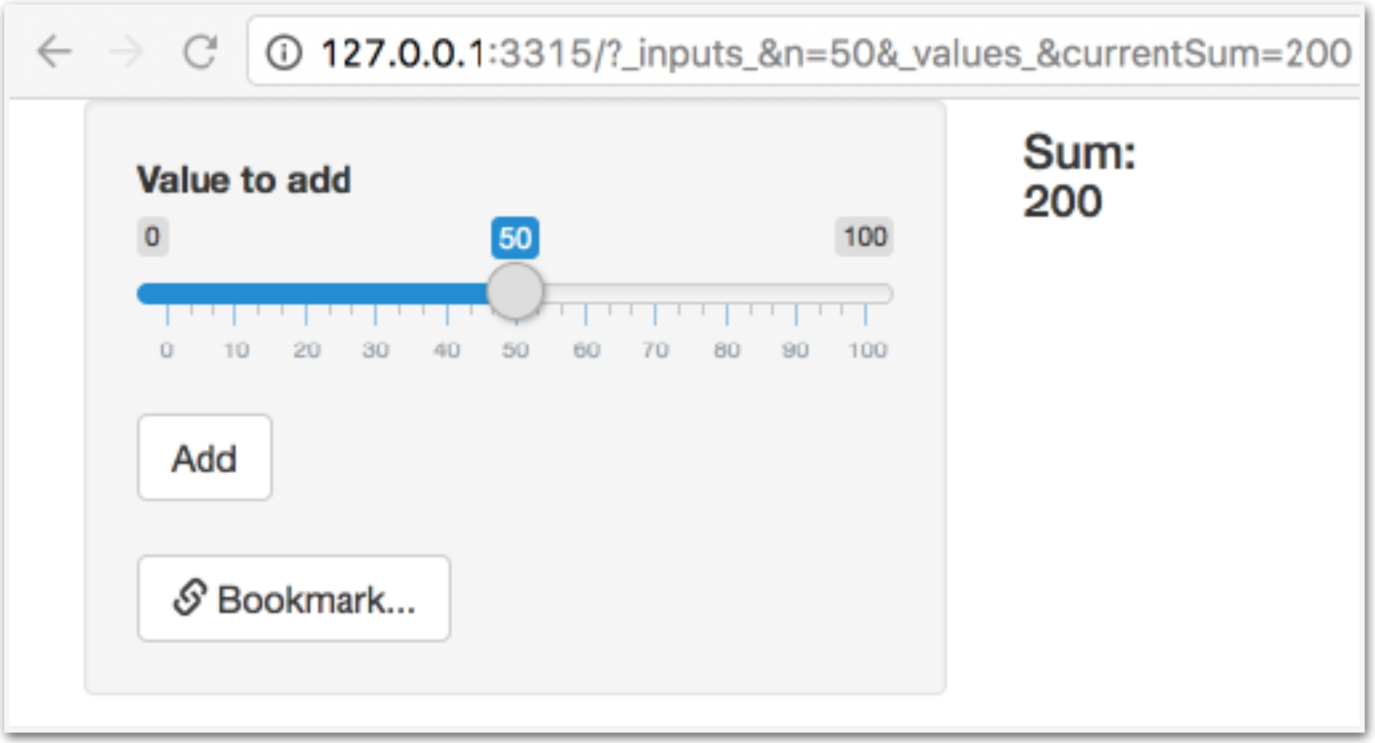


Bookmarked application link

http://127.0.0.1:3315/?_inputs_&n=50&_values_&currentSum=200

This link stores the current state of this application. Press ⌘-C to copy.

Restore


127.0.0.1:3315/?_inputs_&n=50&_values_&currentSum=200

Sum:
200

Advanced bookmarking: useful functions

Callbacks when bookmarking:

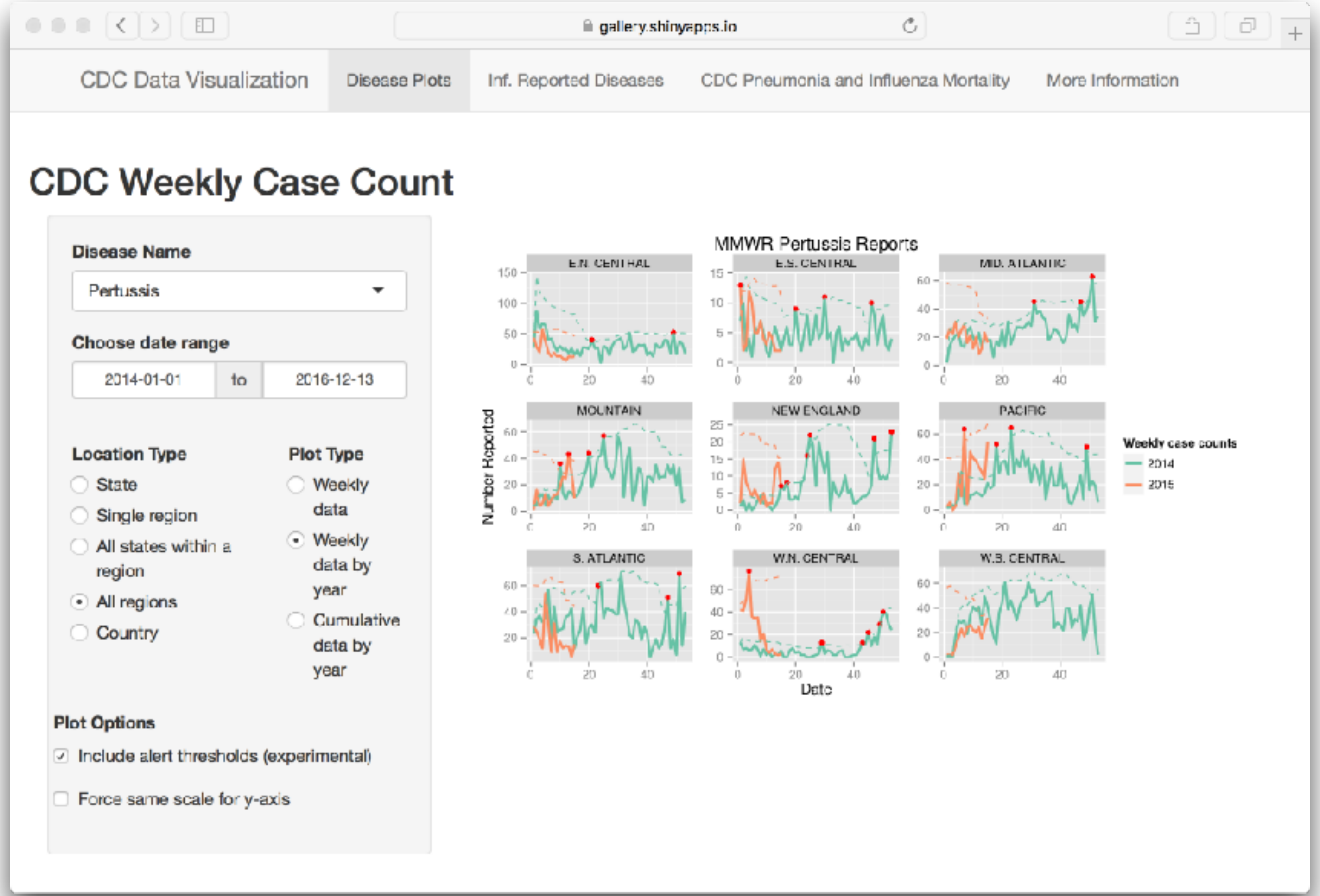
```
onBookmark()
onBookmarked()
```

Callbacks when restoring:

```
onRestore()
onRestored()
```

Exclude values from bookmarking:

```
setBookmarkExclude()
```



CDC Weekly Case Count

Disease Name

Pertussis

Choose date range

2014-01-01 to 2016-12-13

Location Type

○ State
○ Single region
○ All states within a region
● All regions
○ Country

Plot Type

○ Weekly data
● Weekly data by year
○ Cumulative data by year

Plot Options

☑ Include alert thresholds (experimental)

☐ Force same scale for y-axis

https://gallery.shinyapps.io/CDCPlot/

To: Boss <boss@megacorp.com>

Subject: Pertussis data

Hi Boss, here's the latest data:

  https://gallery.shinyapps.io/CDCPlot/?_state_id_=129a0b4c

Enjoy your holidays!

http://shiny.rstudio.com/articles/

Section: Bookmarking State