

USING R SHINY TO CREATE AND EXPAND THUMBNAIL IMAGES

John Hendrickx

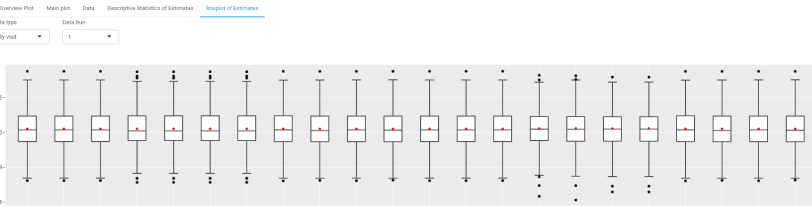
Senior Statistical Programmer

John.Hendrickx@Danone.com



Dynamic Web Pages

Analyse Simulation Estimates Data

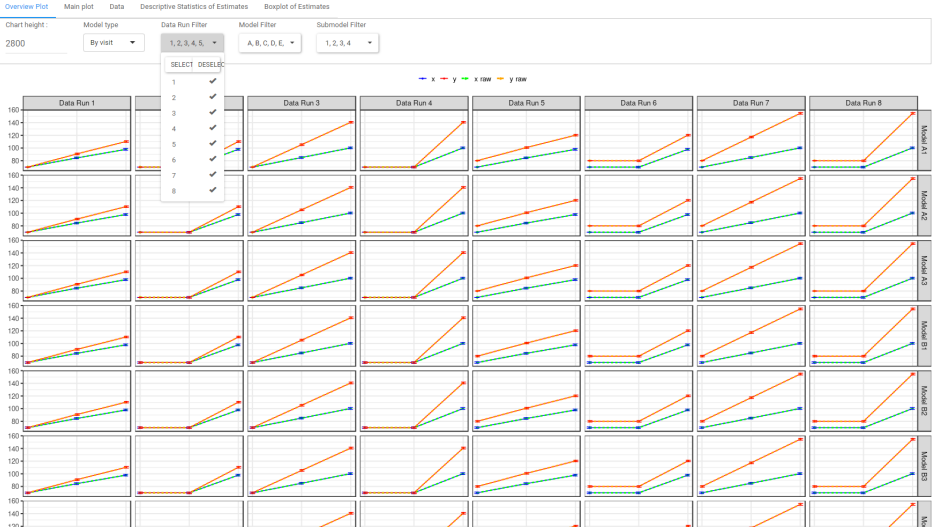


Analyse Simulation Estimates Data

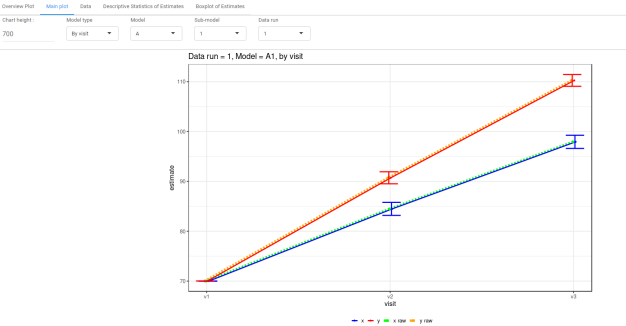
Overview Plot Main plot Data Descriptive Statistics of Estimates Boxplot of Estimates				
Show 10 entries				
	visit	trt	estimate.mean	estimate.sd
1	v1	x	69.9936	0.0000
2	v1	y	70.0114	0.0000
3	v2	x	84.4841	1.3028
4	v2	y	90.7209	1.2064
5	v3	x	97.9295	1.3190
6	v3	y	110.2511	1.1794
7	v1	x raw	69.9936	
8	v1	y raw	70.0114	
9	v2	x raw	84.4804	
10	v2	y raw	90.7197	

Showing 1 to 10 of 12 entries

Previous 1 2 Next



Analyse Simulation Estimates Data



Analyse Simulation Estimates Data

Overview Plot Main plot Data Descriptive Statistics of Estimates Boxplot of Estimates				
Data Run: 1, 2, 3, 4, 5, 6, 7, 8				
Effect: X AT V2				
Model: A, B, C, D, E, F				
statrun	modelcode	label	mean	sd
1	A1	x at v2	84.48	1.30
1	A2	x at v2	84.48	1.30
1	A3	x at v2	84.48	1.30
1	B1	x at v2	84.48	1.34
1	B2	x at v2	84.48	1.34
1	B3	x at v2	84.48	1.34
1	B4	x at v2	84.48	1.34
1	C1	x at v2	84.48	1.34
1	C2	x at v2	84.48	1.34
1	C3	x at v2	84.48	1.34
1	C4	x at v2	84.48	1.34
1	D1	x at v2	84.48	1.30
1	D2	x at v2	83.58	84.49
1	D3	x at v2	84.48	1.30
1	E1	x at v2	84.09	0.97
1	E2	x at v2	84.12	0.97
1	E3	x at v2	84.13	0.97
1	E4	x at v2	84.13	0.97
1	F1	x at v2	84.48	1.34
1	F2	x at v2	84.48	1.34
1	F3	x at v2	84.48	1.34
1	F4	x at v2	84.48	1.34

SELECT ALL Deselect ALL		nobs
CFB x at bl		200
CFB x at v2		200
CFB x at v3		200
CFB y at bl		200
CFB y at v2		200
CFB y at v3		200
CFB y vs x		200
x at bl		200
x at v2		200
x at v3		200
y at bl		200
y at v2		200
y at v3		200
y vs x at v3		200

Expandable Thumbnails

Sensors

- Flat areas, Spikes
- Expand for details

Models

- Which classes fit
- Expand for details

See Github for details

```
library(ggplot2)

# Generate some random data, 3 rows, 4 columns, 10 cases per dataset
df <- expand.grid(x=1:10, rowvar=1:3, colvar=1:4)
df$y <- rnorm(nrow(df))

# Function to create the plots
# If no row or column is specified, a lattice plot is created (thumbnails)
# Otherwise a larger zoomed in plot is created with a title

doPlot <- function(row=NULL, col=NULL) {
  if (is.null(row)|is.null(col)) {usedata <- df}
  else {usedata <- df[df$rowvar==row & df$colvar==col, ]}
  p <- ggplot(usedata, aes(y = y, x=x)) +
    geom_line() +
    labs(y="y")+
    scale_x_discrete(name="x", limits=factor(1:10)) +
    theme_bw(base_size = 18) +
    theme(aspect.ratio = 9/16)
  if (is.null(row)|is.null(col)) {
    p <- p + facet_grid(vars(rowvar), vars(colvar))
  }
  else {
    p <- p + labs(title = paste0("Row ", row, ", column ", col))
  }
  p
}

server <- function(input, output, session) {
  # Initialize -----
  # Initial value to be plotted, updated on click
  selectedRow <- reactiveVal(1)
  selectedCol <- reactiveVal(1)

  observeEvent(input$overviewClick, {
    clickList <- input$overviewClick
    if (!is.null(clickList)) {
      selectedRow(clickList$panelvar2)
      selectedCol(clickList$panelvar1)
      updateTabsetPanel(session, "inTabset", selected = "Expanded version")
    }
  })

  # overview chart =====
  output$overview <- renderPlot({
    doPlot()
  },
  height=600
)

  # Enlarged chart =====
  output$enlarged <- renderPlot({
    doPlot(row=selectedRow(), col=selectedCol())
  },
  height=600
)
}
```



```
library(ggplot2)
library(gridExtra)

# Generate some random data, 10 repeats of x by y, indexed by rep
df <- expand.grid(x=1:10, reps=1:12)
df$y <- rnorm(nrow(df))

doPlot <- function(forvalue) {
  ggplot(df[df$reps==forvalue,], aes(y = y, x=x)) +
    geom_line() +
    labs(y="y")+
    labs(title = paste0("value ", forvalue))+
    scale_x_discrete(name="x", limits=factor(1:10)) +
    theme_bw(base_size = 18) +
    theme(aspect.ratio = 9/16)
}

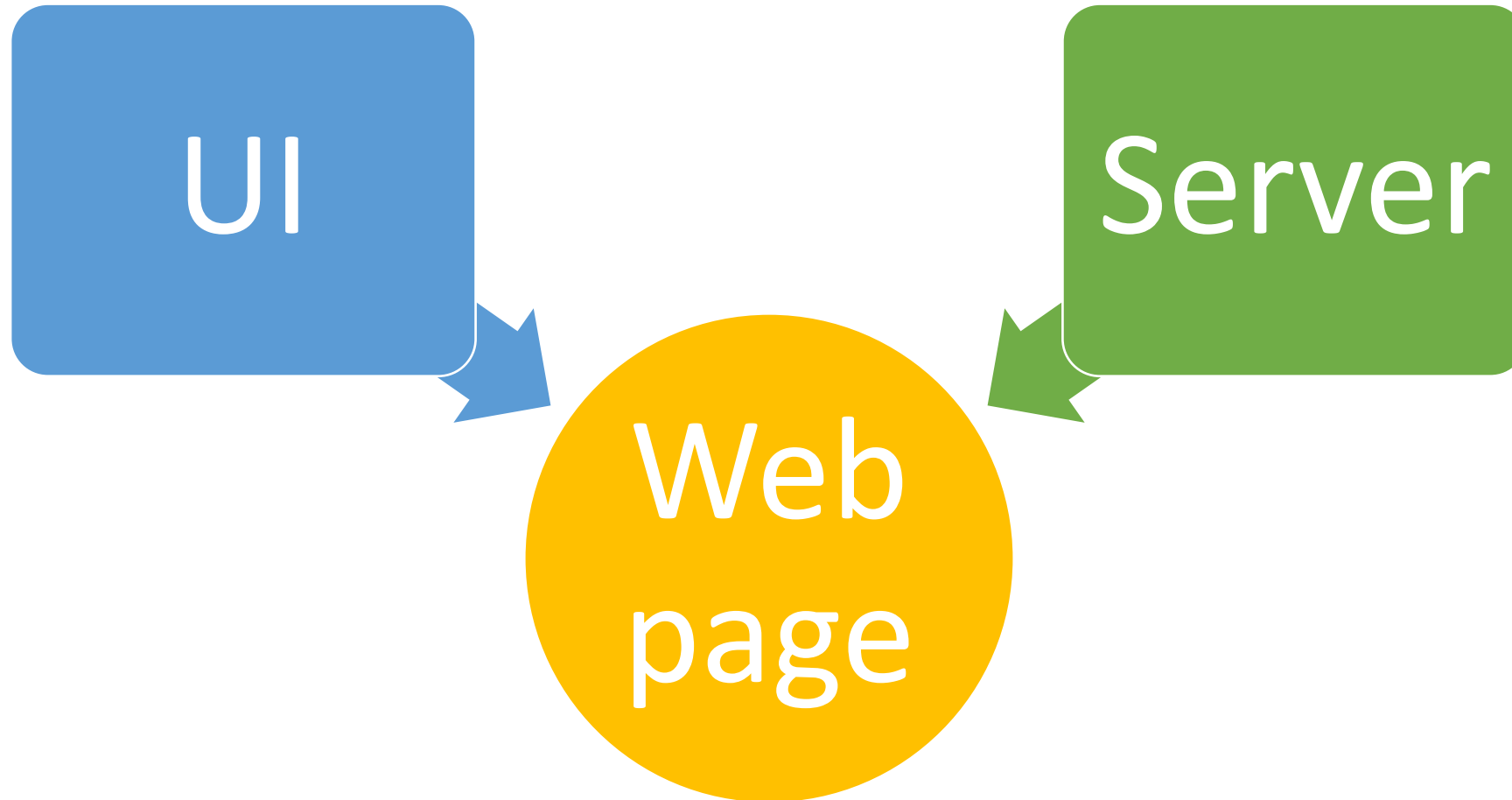
server <- function(input, output, session) {
  # Initialize -----
  # Initial value to be plotted, updated on doubleclick
  plotThis <- reactiveVal(1)

  observeEvent(input$overviewClick, {
    clickList <- input$overviewClick
    if (!is.null(clickList)) {
      ncol <- 4
      nrow <- ceiling(length(unique(df$reps))/ncol)
      chartCol <- ceiling(clickList$x*ncol)
      chartRow <- ceiling((1 - clickList$y)*nrow)
      plotThis((chartRow-1)*ncol+chartCol)
      # cat(nrow,"rows", ncol,"columns\n", "Row", chartRow,"column", chartCol,"n")
      updateTabsetPanel(session, "inTabset", selected = "Expanded version")
    }
  })

  # overview chart =====
  output$overview <- renderPlot({
    a <- lapply(unique(df$reps), function(val) {return(doPlot(forvalue = val))})
    grid.arrange(grobs=a, ncol=4)
  },
  height=600
)

  # Enlarged chart =====
  output$enlarged <- renderPlot({
    doPlot(forValue = plotThis())
  },
  height=600
)
}
```

Shiny



UI

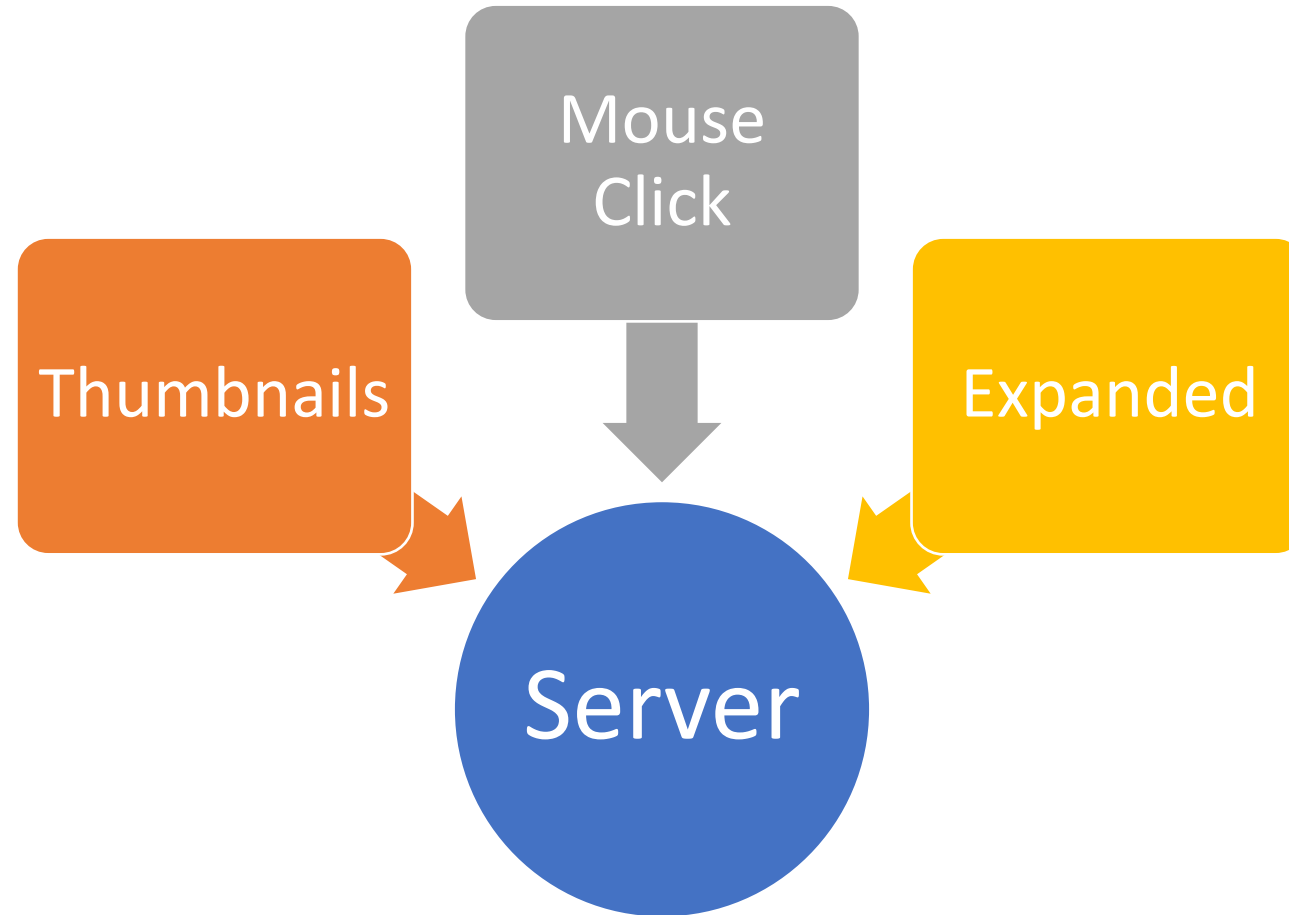
Thumbnail tab

- Thumbnail charts
- Click event

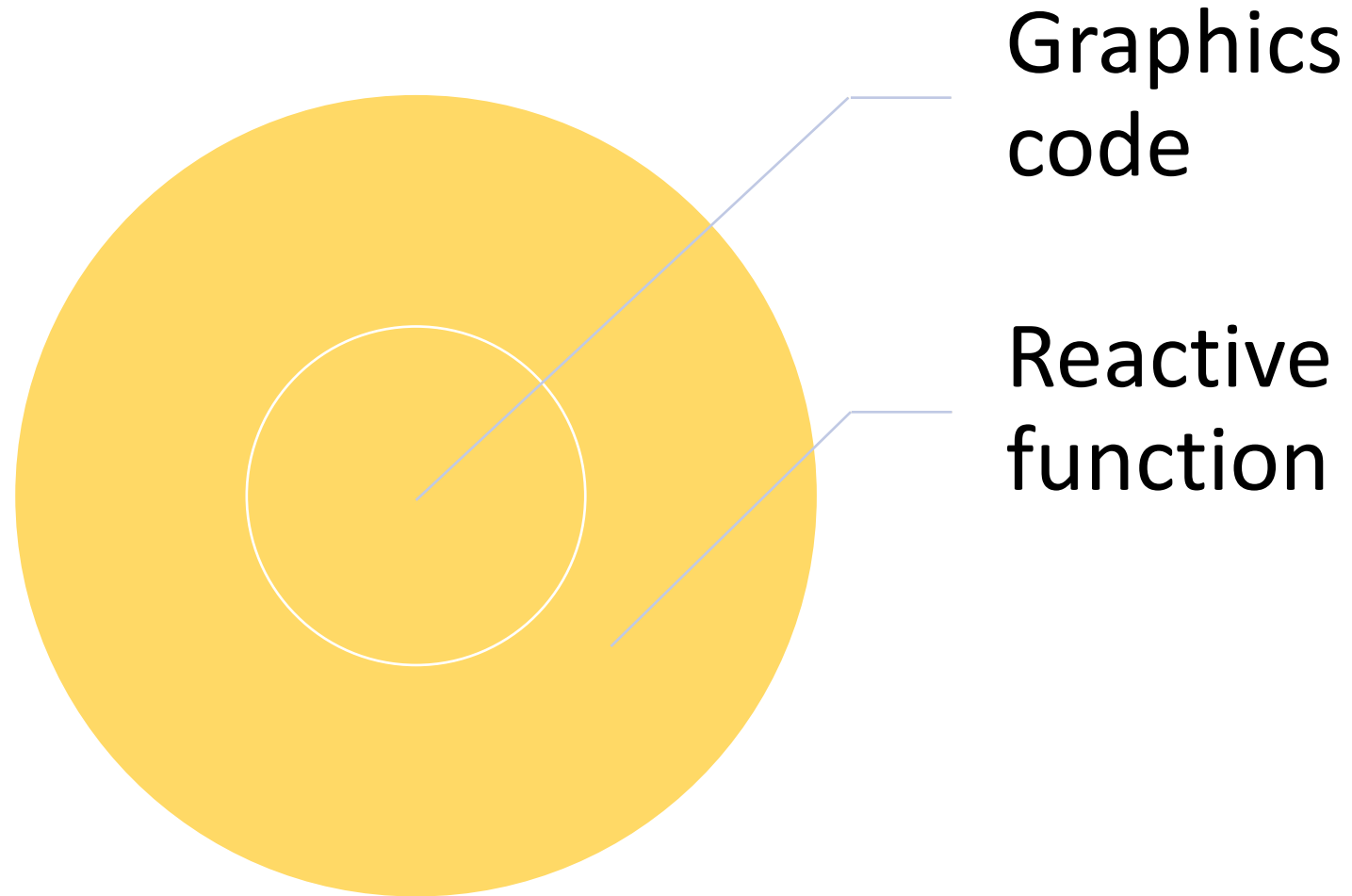
Expanded tab

- Chart

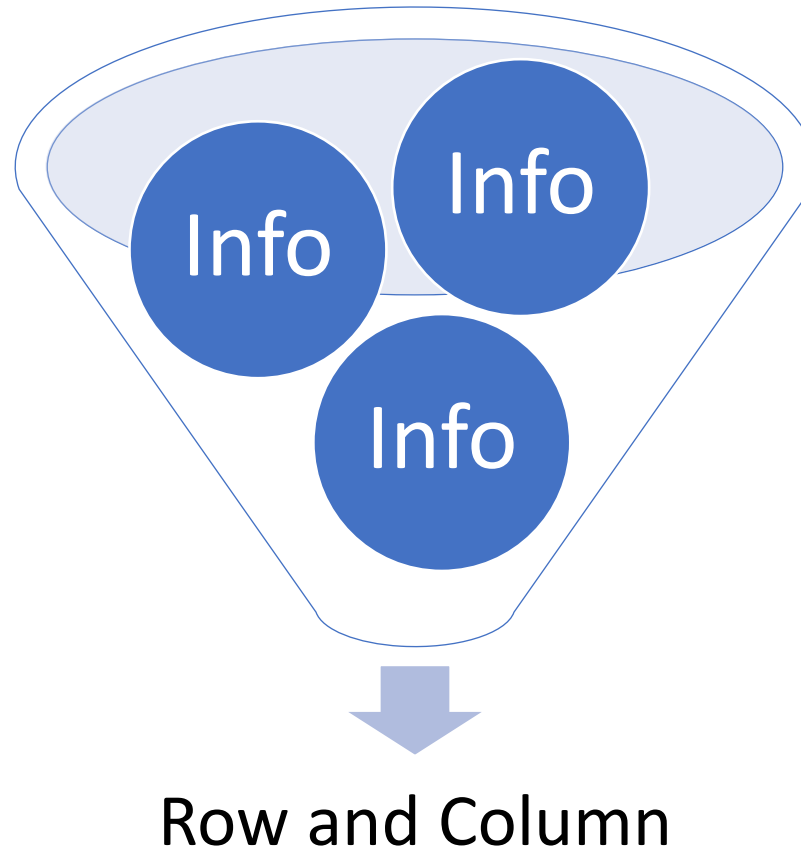
Server



Thumbnails



Mouse event



reactiveVal

```
selectedRow <- reactiveVal(1)
```

- Assign value 1

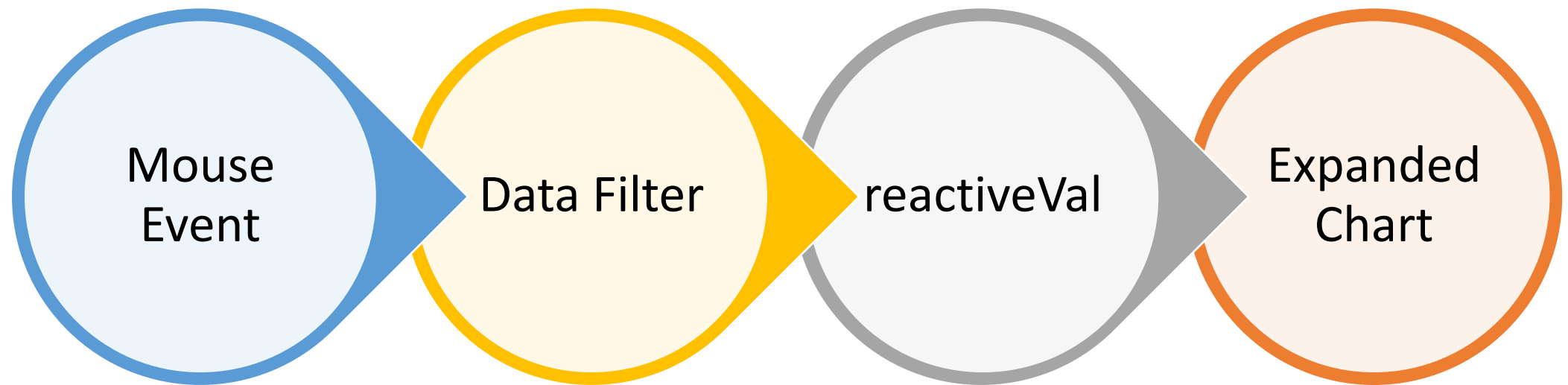
```
selectedRow()
```

- Retrieve current value

```
selectedRow(clickList$panelvar2)
```

- Assign row

ReactiveVal



Thumbnail plots

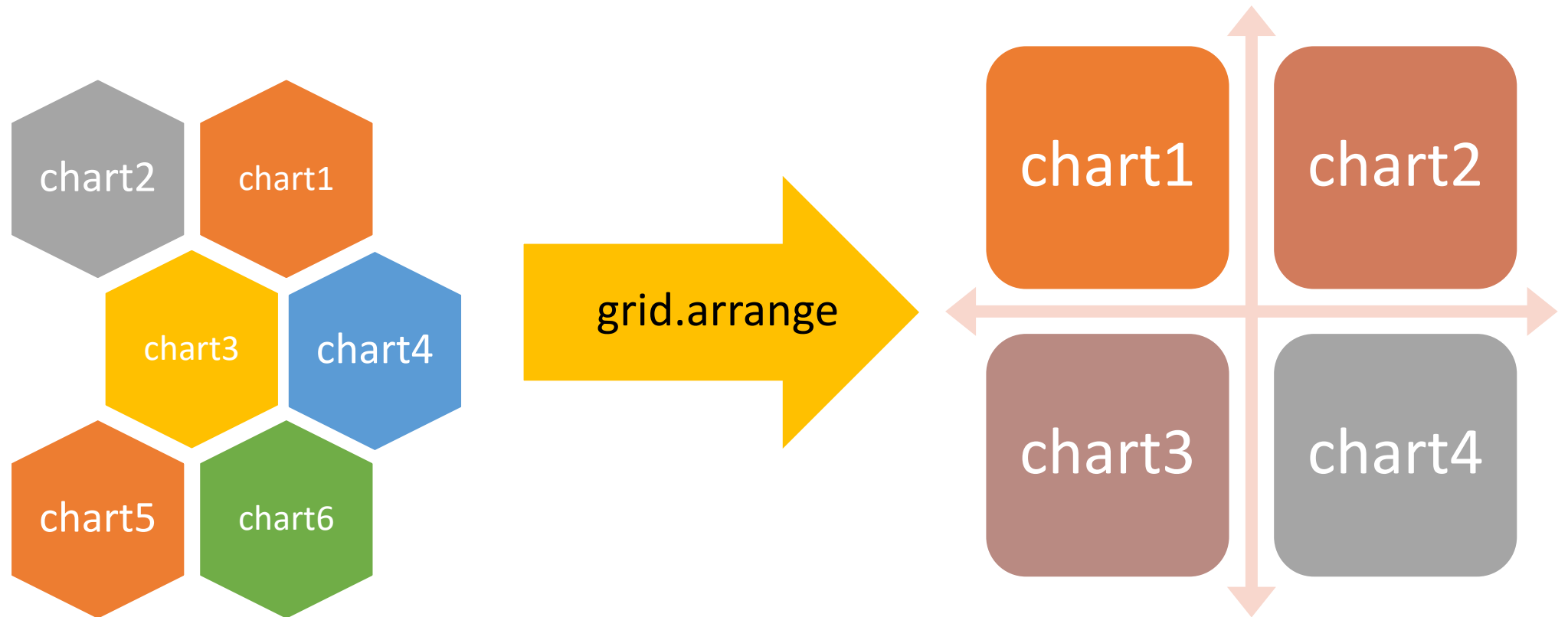
Lattice
plot

- Row and column variables
- Click contains filters

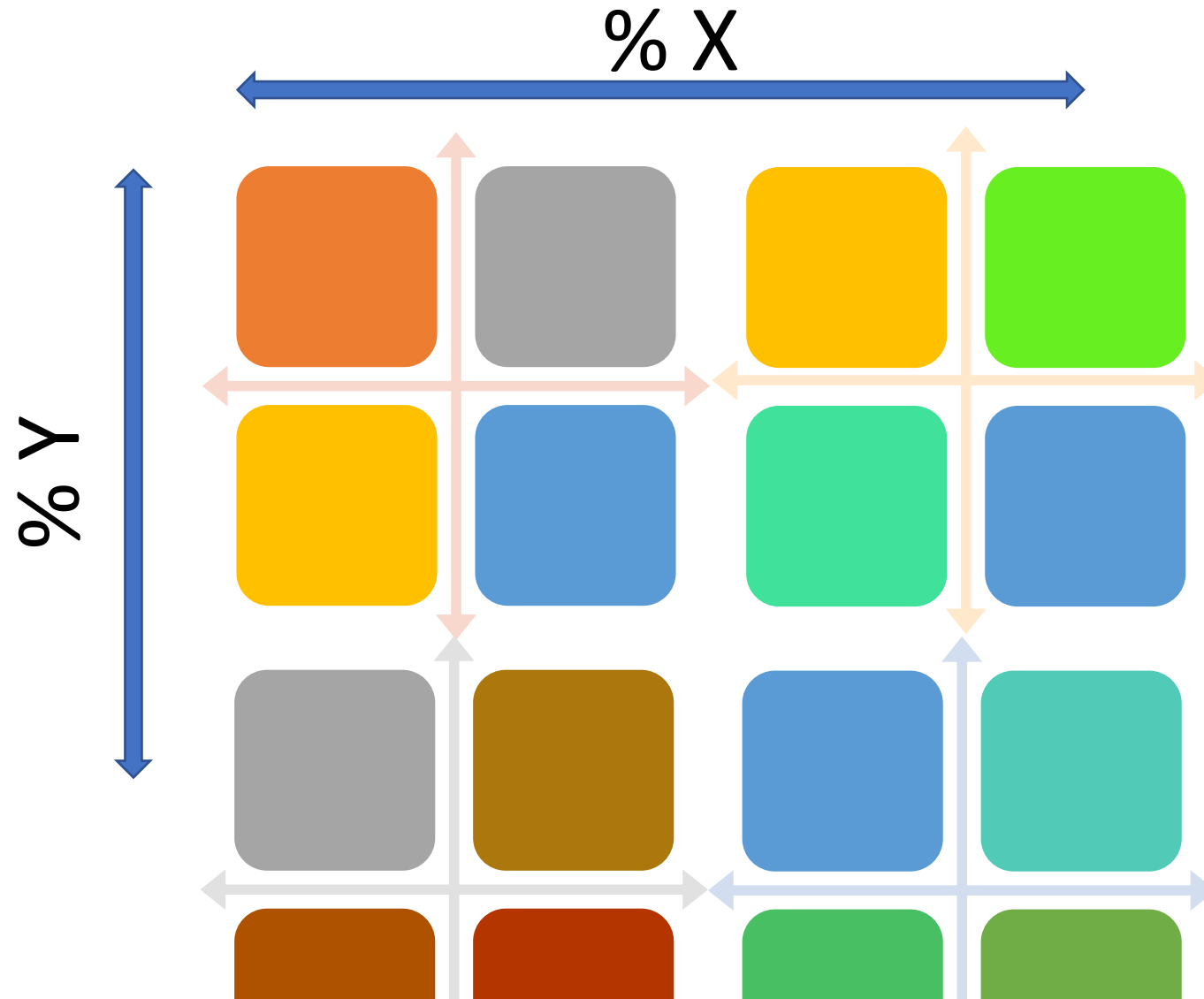
List of
charts

- Derive filters
- Grid of charts

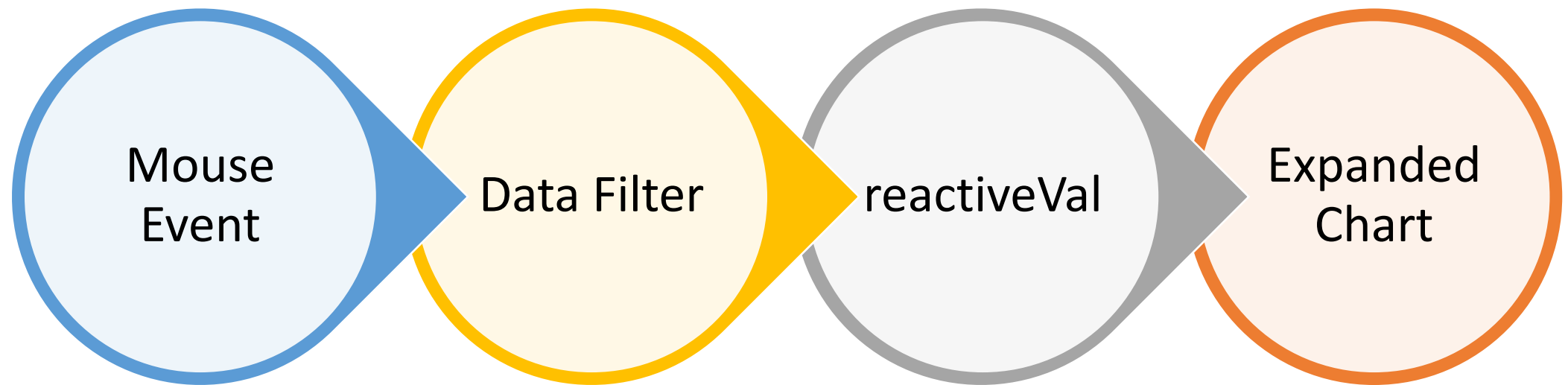
List of charts



Mouse event II



ReactiveVal



In Summary

Web page with two tabs

Click event for thumbnails

Create thumbnails

Process click

Pass on data filter (reactiveVal)

Create expanded

Enhancements

Filtering

Button for thumbnails

Hover info

Chart data

Summarise data

Github page

- Programs at <https://github.com/JohnHendrickx/Phuse2020>
- Run examples from RStudio
`runGitHub("JohnHendrickx/Phuse2020",
 username="JohnHendrickx",
 subdir="Example 2")`
 - Requires packages “shiny”, ggplot2, gridExtra

