An Example of "Linking" Graphical Objects in **grid**

Paul Murrell

November 14, 2026

Suppose that I am drawing two graphs on a page, which are the results from two subjects in an experiment. I want the graphs to have the same axes to aid in comparison of the subjects.

First of all, I will split the page up into two bits for the two graphs.

```
> pushViewport(viewport(layout = grid.layout(1, 2, respect = TRUE)))
```

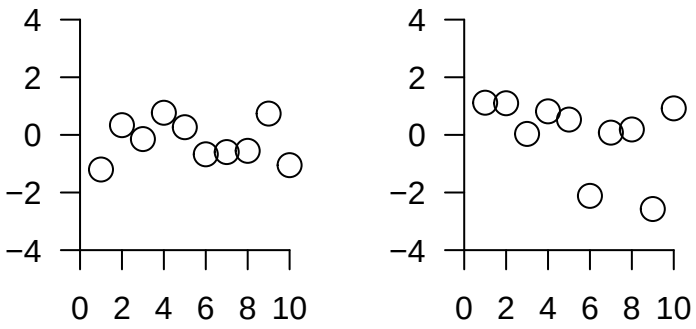
Now I generate some data and draw the first plot.

```
> x <- 1:10
> y1 <- rnorm(10)
> vp1a <- viewport(layout.pos.col = 1)
> vp1b <- viewport(width = 0.6, height = 0.6,
+                  xscale = c(0, 11), yscale = c(-4, 4))
> pushViewport(vp1a, vp1b)
> grid.xaxis(name = "xaxis")
> grid.yaxis(name = "yaxis")
> grid.points(x, y1)
> popViewport(2)
```

Notice that I have named the graphical objects for the axes. When I draw the second plot I will use the same names for the axis objects.

```
> y2 <- rnorm(10)
> vp2a <- viewport(layout.pos.col = 2)
> vp2b <- viewport(width = 0.6, height = 0.6,
+                  xscale = c(0, 11), yscale = c(-4, 4))
> pushViewport(vp2a, vp2b)
> grid.xaxis
> grid.xaxis(name = "xaxis")
> grid.yaxis(name = "yaxis")
> grid.points(x, y2)
> popViewport(2)
```

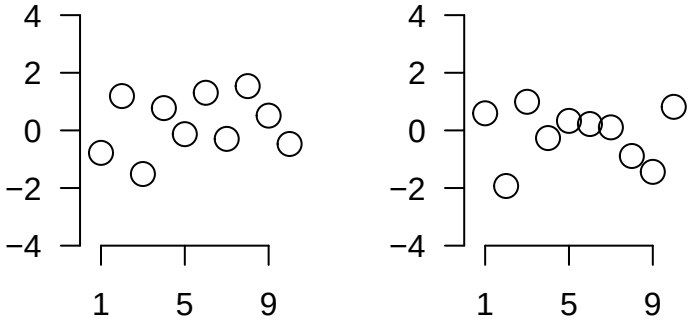
The output looks like the figure below.



Because I have used the same name for the axis objects in both plots, I can edit the axes for both plots simultaneously rather than having to edit each one in turn. For example ...

```
> grid.edit("xaxis", at = c(1, 5, 9), global = TRUE)
>
```

The output now looks like the figure below.



This might seem a very small gain in this example, but it is potentially of great use in, for example, the implementation of a scatterplot matrix.