Saving and Loading **grid** Graphics

Paul Murrell

October 31, 2026

This is a general discussion concerning how you might go about creating (and reusing) persistent representations of **grid** graphics, and some of the pitfalls in the various options.

R code

The way I usually work with graphics is to write R code in a text file and copy-and-paste or `source()` it into R. In this case, the persistent representation of the graphics is the raw R code.

The representation is fully editable.

The representation is persistent across R sessions, but may not be persistent across R versions because the names, argument lists, and/or behaviour of the **grid** functions may change. The representation can be reloaded into R. Incompatibilities between versions should be handled gracefully by R's argument-matching, type-checking, and/or version-checking.

grid `grobs`

The next most flexible way of creating a persistent version of **grid** graphics is to `save()` a **grid** `grob` ...

```
> gt <- textGrob("hi")
> save(gt, file = "mygridplot")
```

The representation is reloadable so you can reproduce an image ...

```
> load("mygridplot")
> grid.draw(gt)
```

And the representation is editable; there is an API for interacting with **grid** `grobs`, including adding new elements, removing elements, editing features of a `grob` and so on. See `getGrob()`, `addGrob()`, `removeGrob()`, and `editGrob()`.

The representation is persistent across R sessions, but is vulnerable to changes in the internal representation of **grid** `grobs`. The advantage of this representation is that it is possible to

share and edit graphics produced by someone else, without seeing the code they used to produce it.

Device output

A third way of creating a persistent version of **grid** graphics is to "save" it to a persistent device format (e.g., PostScript, PDF, ...).

It is possible to edit these representations, but hardly convenient. In particular, none of the coordinate systems used in the construction of plots (e.g., axis scales) are available for editing.

The representation is persistent regardless of R sessions or versions, but it cannot be reloaded into R.

Display lists

This final option is *not* recommended, but it is possible to do so its downsides need to be explained.

What you can do is "save" an R display list using, for example, ...

```
> grid.grill()
> temp <- recordPlot()
> save(temp, file = "mygridplot")
```

This representation is not editable[1], but it can be reloaded and rerun to reproduce the output ...

```
> load("mygridplot")
> temp
```

The representation is persistent across R sessions, but the saved information involves non-public interfaces and structures which may change between R versions. You also have to make sure that **grid** has been reloaded. Differences between **grid** and/or R versions may lead to segmentation faults.

[1] Well, there's nothing stopping you editing it, but you should take out life insurance first.