

Saving and Loading `grid` Graphics

Paul Murrell

July 9, 2003

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.

Device output

Another 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 this representation, but hardly convenient.

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

Display lists

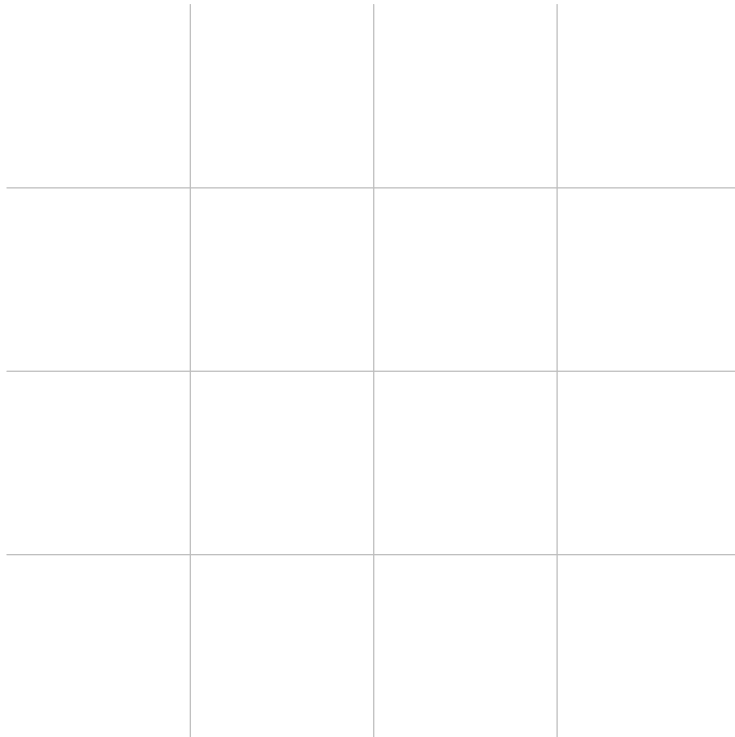
A third way of creating a persistent version of `grid` graphics is to “save” an R display list using, for example, ...

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

This representation is not editable¹, but it can be reloaded ...

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

... and rerun to reproduce the output.



The representation is persistent across R sessions (although you have to make sure that `grid` has been reloaded). Differences between `grid` versions may lead to segmentation faults if the API of the `grid` code being called by the display list has changed sufficiently.

grid grobs

You might think that yet another way of creating a persistent version of `grid` graphics is to `save()` a `grid grob`. It is true that something like the following will run ...

```
> gt <- grid.text("hi")  
> save(gt, file = "mygridplot")
```

¹Well, there's nothing stopping you editing it, but you should take out life insurance first. Display lists may become sensibly editable in the future.

... but the saved object is empty²

The representation is reloadable ...

```
> load("mygridplot")
> unclass(gt)
```

```
[[1]]
<pointer: (nil)>
```

... but the resulting object is empty.

The representation is persistent (if useless) across R sessions, and is indifferent(!) to differences between versions of R and `grid`.

The behaviour of this sort of representation may change in the future with the advent of namespaces.

²A `grid grob` contains an external C pointer to an R object (to allow unique references to `grid` objects). The `save()` code sets such external pointers to `NULL`.