Inserting references in Rd and roxygen2 documentation

Georgi N. Boshnakov
University of Manchester

Abstract

Package **Rdpack** provides a straightforward way to insert BibTeX references in the documentation of an R package, using the RdMacros feature introduced in R 3.2.0. It works for 'roxygen2' documentation, as well.

This vignette is part of package Rdpack, version 0.6-1.

Keywords: bibliographic references, Rd, bibtex, R.

1. Setup

To prepare a package for importing BibTeX references it is necessary to tell the package management tools that package **Rdpack** and its Rd macros are needed. The references should be put in file `inst/REFERENCES.bib`. These steps are enumerated below in somewhat more detail for convenince:

1. Add the following lines to file 'DESCRIPTION':

```
Imports: Rdpack
RdMacros: Rdpack
```

Make sure the capitalisation of `RdMacros` is as shown. If the field 'RdMacros' is already present, add 'Rdpack' to the list on that line. Similarly for field 'Imports'.

2. Add the following line to file 'NAMESPACE'[1]:

```
importFrom(Rdpack,reprompt)
```

The equivalent line for 'roxygen2' is

```
#' @importFrom Rdpack reprompt
```

3. Create file `REFERENCES.bib` in subdirectory `inst/` of your package and put the bibtex references in it.

2. Inserting references in package documentation

Once the steps outlined in the previous section are done, references can be inserted in the documentation as `\insertRef{key}{package}`, where `key` is the bibtex key of the reference and `package` is your package. This works in Rd files and in roxygen documentation chunks.

[1] Any function for package **Rdpack** will do. This is to avoid getting a warning from 'R CMD check'.

In fact, argument 'package' can be any installed R package[2], not necessarily the current one. This means that you don't need to copy references from other packages to your `"REFERENCES.bib"` file. This works for packages that have `"REFERENCES.bib"` in their installation directory and for the default packages.

See also the help pages `?Rdpack::insertRef` and `?Rdpack::Rdpack-package`. For example, the help page `?Rdpack::insertRef` contains the following lines in section "References" of the Rd file:

```
\insertRef{Rpack:bibtex}{Rdpack}
\insertRef{R}{bibtex}
```

The first line above inserts the reference labeled `Rpack:bibtex` in Rdpack's `REFERENCES.bib`. The second line inserts the reference labeled `R` in file `REFERENCES.bib` in package 'bibtex'.

A roxygen2 documentation chunk might look like this:

```
#' @references
#' \insertRef{Rpack:bibtex}{Rdpack}
#'
#' \insertRef{R}{bibtex}
```

3. Inserting citations

From version 0.6-1 of **Rdpack**, additional Rd macros are available for citations. They can be used in both Rd and roxygen2 documentation.

`\insertCite{key}{package}` cites the key and records it for use by `\insertAllCited`, see below. `key` can contain more keys separated by commas.

Documentation source	rendered
`\insertCite{parseRd,Rpack:bibtex}{Rdpack}`	(Murdoch 2010; Francois 2014)
`\insertCite{Rpack:bibtex}{Rdpack}`	(Francois 2014)

By default the citations are parenthesised `\insertCite{parseRd}{Rdpack}` produces (Murdoch 2010). To get textual citations, like Murdoch (2010) put the string `;textual` at the end of the key. The references in the last two sentences would be produced with `\insertCite{parseRd}{Rdpack}` and `\insertCite{parseRd;textual}{Rdpack}`, respectively. This also works with several citations, e.g. `\insertCite{parseRd,Rpack:bibtex;textual}{Rdpack}` produces: Murdoch (2010); Francois (2014).

The macro `\insertNoCite{key}{package}` records one or more references for `\insertAllCited` but does not cite it. Setting `key` to `*` will include all references from the specified package. For example, `\insertNoCite{R}{bibtex}` and `\insertNoCite{*}{utils}` record the specified references for inclusion by `\insertAllCited`.

`\insertAllCited` inserts all references cited with `\insertCite` or `\insertNoCite`. Putting this macro in the references section will keep it up to date automatically. The Rd section may look something like:

```
\insertAllCited
```

or, in roxygen2, the references chunk might look like this:

```
#' @references
#'     \insertAllCited
```

[2]There is of course the risk that the referenced entry may be removed from the other package. So this is probably only useful for one's own packages. Also, the other package would better be one of the packages mentioned in DESCRIPTION.

To mix the citations with other text, such as "see also" and "chapter 3", write the list of keys as a free text, starting it with the symbol `@` and prefixing each key with it. The `@` symbol will not appear in the output. For example, the following code

```
\insertCite{@see also @parseRd and @Rpack:bibtex}{Rdpack}
\insertCite{@see also @parseRd; @Rpack:bibtex}{Rdpack}
\insertCite{@see also @parseRd and @Rpack:bibtex;textual}{Rdpack}
```

produces:

(see also Murdoch 2010 and Francois 2014)
(see also Murdoch 2010; Francois 2014)
(see also Murdoch (2010) and Francois (2014))

`\insertCiteOnly{key}{package}` is as `\insertCite` but does not include the key in the list of references for `\insertAllCited`.

4. Possible issues

4.1. Warning from 'R CMD build'

If 'R CMD build' or `devtools::build()` gives a warning along the lines of:

```
Warning: C:/temp/RtmpqWQqji/.../XXX.Rd:52: unknown macro '\insertRef'
```

then check the syntax in file DESCRIPTION — the most common cause of this is misspelling `RdMacros:`. Make sure in particular that 'M' is uppercase.

4.2. Development using 'devtools':

The described procedure works transparently in 'roxygen2' chunks and with Hadley Wickham's 'devtools'. Packages are built and installed properly with the 'devtools' commands and the references are processed as expected.

Currently (2017-08-04) if you run help commands `?xxx` for functions from the package you are working on and their help pages contain references, you may encounter some puzzling warning messages in 'developer' mode, something like:

```
1: In tools::parse_Rd(path) :
  ~/mypackage/man/abcde.Rd: 67: unknown macro '\insertRef'
```

These warnings are again about unknown macros but the reason is completely different: they pop up because "devtools" reroutes the help command to process the developer's Rd sources (rather than the documentation in the installed directory) but doesn't tell `parse_Rd` where to look for additional macros[3].

These warnings are harmless - the help pages are built properly and no warnings appear outside "developer" mode, e.g. in a separate R session. See below for a way to inspect help pages directly from Rd files.

4.3. Viewing Rd files

[3]The claims in this sentence can be deduced entirely from the informative message. Indeed, (1) the error is in processing a source Rd file in the development directory of the package, and (2) the call to `parse_Rd` specifies only the file.

A function, `viewRd`, to view Rd files in the source directory of a package was introduced in version 0.4-23 of **Rdpack**. A typical user call would look something like:

```
Rdpack::viewRd("./man/filename.Rd")
```

By default the requested help page is shown in text format. To open the page in a browser, set argument `type` to `"html"`:

```
Rdpack::viewRd("./man/filename.Rd", type = "html")
```

Users of 'devtools' can use `viewRd()` in place of `help()` to view Rd sources[4].

5. Inserting references interactively

It is possible to use the underlying R function to insert references interactively. For example,

```
> library(Rdpack)
> cat(insert_ref("R", package = "bibtex"), sep ="\n")

R Development Core Team (2009).
\emph{R: A Language and Environment for Statistical Computing}.
R Foundation for Statistical Computing, Vienna, Austria.
ISBN 3-900051-07-0, \url{https://www.R-project.org}.
```

I would put the (commented out) command on top of the above reference as a reminder where it came from:

```
% insert_ref("R", package = "bibtex"), sep ="\n")
```

For a different approach, see the documentation of function `{Rdpack::rebib()}`.

Affiliation:

Georgi N. Boshnakov
School of Mathematics
The University of Manchester
Oxford Road, Manchester M13 9PL, UK
URL: http://www.maths.manchester.ac.uk/~gb/

[4] Yes, your real sources are the `*.R` files but `devtools::document()` transfers the roxygen2 documentation chunks to Rd files (and a few others), which are then rendered by **R** tools.