Inserting references in Rd and roxygen2 documentation

Georgi N. Boshnakov

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.

To use this mechanism in your package:

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

   ```
   RdMacros: Rdpack
   ```

   (If the field 'RdMacros' is already present, add Rdpack to the list on that line.)

2. Add 'Rdpack' to the list of imports[1]. If you do not have one, add the following line to file 'DESCRIPTION':

   ```
   Imports: Rdpack
   ```

   You will need also to import something from the package, e.g. put this in file 'NAMESPACE':

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

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

Then you can insert references in the documentation with `\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] Currently 'R CMD check' doesn't complain if you don't import 'Rdpack', as long as 'Rdpack' is installed on the system (yours, CRAN, win-builder, etc.). However, services like 'appveyor' or 'travis-ci' do raise errors. Thanks to Clemens Schmid and Tim Riffe for allerting me about this.

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}
```

Note about 'devtools':

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

Currently (2017-08-04) you may encounter some puzzling warning messages in 'developer' mode, if you run help commands `?xxx` for functions from the package you are working on and their help pages contain references. The warnings 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` were to look for additional macros.

These warnings are harmless - the help pages are built properly and no warnings appear outside "developer" mode, e.g. in a separate R session.

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")
```

[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.

```
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()}`.