Our website is based on Markdown content rendered with Hugo..  
In particular, we cannot use the usual BibTex + CSL + Pandoc-citeproc to handle a bibliography.  
Thankfully, using the rOpenSci package RefManageR, we can still make our own bibliography from a BibTeX file without formatting references by hand.  
In this post we shall present our custom workflow for inserting citations, as well as more mainstream tools.

**Usual citation workflow in Rmd**

To repeat information presented in [Nicholas Tierney’s excellent online book “R Markdown for Scientists”](https://rmd4sci.njtierney.com/citing-articles-bibliography-styles.html), to get a bibliography we have to perform several steps. We must…

* reference a bibliography style in the document YAML metadata, with entries such as the one below for R,

@Manual{my-citation-key-for-r,

title = {R: A Language and Environment for Statistical Computing},

author = {{R Core Team}},

organization = {R Foundation for Statistical Computing},

address = {Vienna, Austria},

year = {2020},

url = {<https://www.R-project.org/>},

}

* insert citations using their key, e.g. @my-citation-key-for-r for the entry above,
* and, optionally, use CSL to control the look of the references list; and a bookdown output format to have the bibliography placed somewhere else than the end.

**Handy R packages for references: citr and knitcitations**

Some R tools out there simplify part of the workflow: **citr and knitcitations**

* citr provides an RStudio add-in to search for references in a .bib file, inserting the key in the document (mimicking tools you could find in Microsoft Word);
* **knitcitations** allows you to not only use the keys, but also DOIs and URLs to cite a paper. E.g. citet("10.1098/rspb.2013.1372") will create a citation to @Boettiger\_2013 after querying the web; and write.bibtex(file = "references.bib") will allow you to cache entries in a bibliography file.

Both these packages import RefManagerR, a package that *provides tools for importing and working with bibliographic references*,.

**Citation workflow without Pandoc, with RefManageR**

Thankfully, using RefManageR, we can provide a workflow for adding citations from a bib file, by creating functions similar to knitcitations’ functions.

In the following examples, we’ll use the .bib file below,

@Manual{my-citation-key-for-r,

title = {R: A Language and Environment for Statistical Computing},

author = {{R Core Team}},

organization = {R Foundation for Statistical Computing},

address = {Vienna, Austria},

year = {2020},

url = {<https://www.R-project.org/>},

}

@Article{refmanager,

author = {Mathew William McLean},

title = {RefManageR: Import and Manage BibTeX and BibLaTeX References in R},

journal = {The Journal of Open Source Software},

year = {2017},

doi = {10.21105/joss.00338},

}

@Manual{jqr,

title = {jqr: Client for 'jq', a 'JSON' Processor},

author = {Rich FitzJohn and Jeroen Ooms and Scott Chamberlain and {Stefan Milton Bache}},

year = {2018},

note = {R package version 1.1.0},

url = {<https://CRAN.R-project.org/package=jqr>},

}

**Create a bibliography object**

We’ll start by creating a BibEntry object by calling RefManageR::ReadBib():

mybib <- RefManageR::ReadBib("refs.bib", check = FALSE)

mybib

[1] R. FitzJohn, J. Ooms, S. Chamberlain, et al. \_jqr: Client for 'jq',

a 'JSON' Processor\_. R package version 1.1.0. 2018. .

[2] M. W. McLean. "RefManageR: Import and Manage BibTeX and BibLaTeX

References in R". In: \_The Journal of Open Source Software\_ (2017).

DOI: 10.21105/joss.00338.

[3] R Core Team. \_R: A Language and Environment for Statistical

Computing\_. R Foundation for Statistical Computing. Vienna, Austria,

2020. .

class(mybib)

[1] "BibEntry" "bibentry"

**Create and use a cite function**

Then in another R code chunk, we’ll create a function which writes the citations in the format we want. Our citation format is a footnote: [^key].

cite <- function(key, bib = mybib) {

# create a silent reference so RefManageR

# knows we've used this entry

RefManageR::NoCite(bib, key)

# create the string we need in the Markdown document

paste0("[^", key, "]")

}

From the three entries in the BibTeX file, let us cite two of them.

In the Rmd file we write:

what a nice package`r cite("refmanager")` built on top of R`r cite("my-citation-key-for-r")`.

Which when knit, becomes this in the md file:

what a nice package[^refmanager] built on top of R[^my-citation-key-for-r].

Which when rendered by Hugo, becomes this in the html file: what a nice package[1](https://ropensci.org/technotes/2020/05/07/rmd-citations/#fn:1) built on top of R[2](https://ropensci.org/technotes/2020/05/07/rmd-citations/#fn:2).

**Print the references list**

Finally, for the bibliography to appear, we need to add a call to RefManageR::PrintBibliography(), after defining our own bibstyle.  
This way, each entry will appear in the Markdown file as

[^mykey]:

To get the first part ([^mykey]), we use the fmtPrefix of tools::bibstyle().  
To get the second part (), we read through some code as attached in this repository, in which we found how to create a function that correctly adds DOIs and URLs.  
Note that in our case the order of appearance of entries within the references list in the Markdown file is not important at all, since the Hugo Markdown handler, Goldmark, will reorder them based on order of appearance in the text.

To produce the bibliography, we combined everything together in an R code chunk. For example, the chunk below was used at the very end of the R Markdown file producing this post.  
It came before the definition of another, non-citation, footnote.[3](https://ropensci.org/technotes/2020/05/07/rmd-citations/#fn:3)

```{r bib, echo = FALSE, results = "asis"}

get\_doi <- function(paper) {

if (is.null(paper$doi)) {

""

} else{

paste0(" , https://doi.org/", paper$doi)

}

}

apa\_style <- tools::bibstyle("apa", sortKeys = function(refs) seq\_along(refs),

fmtPrefix = function(paper) paste0("[^", attr(paper,"key"), "]:"),

extraInfo = function(paper) paste0(paper$url, get\_doi(paper)),

.init = TRUE)

RefManageR::PrintBibliography(mybib, .opts = list(bib.style = "apa", sorting = ""))

```

This produces the following in markdown, which is then used by Hugo to produce the citations for this post.

[^refmanager]: McLean MW (2017). "RefManageR: Import and Manage BibTeX

and BibLaTeX References in R." \_The Journal of Open Source Software\_.

, <https://doi.org/10.21105/joss.00338>.

[^my-citation-key-for-r]: R Core Team (2020). \_R: A Language and

Environment for Statistical Computing\_. R Foundation for Statistical

Computing, Vienna, Austria. <https://www.R-project.org/>.

**Workflow summary**

To summarize our workflow

* Instead of referencing the bibliography file in the Document metadata we create an object referring to it in an R code chunk by calling RefManageR::ReadBib();
* We use a custom-made cite() function using RefManageR::NoCite() to signal the use of the reference, and string manipulation to add it in the correct format;
* We make the bibliography appear using RefManageR::PrintBibliography and tools::bibstyle() in a chunk with the results="asis" option. We put that chunk exactly where we want the bibliography to appear.

**Do even more with BibTeX files**

The workflow presented earlier was a cool example of using RefManageR to use a BibTeX file.  
There are two other packages that are worth knowing about for more .bib gymnastics: bib2df and handlr.

**bib2df: from BibTeX to tibble**

bib2df is a package that converts bibliography data from .bib to tibble and vice versa.

Let’s try it on our .bib file.

df <- bib2df::bib2df("refs.bib")

df

# A tibble: 3 x 28

CATEGORY BIBTEXKEY ADDRESS ANNOTE AUTHOR BOOKTITLE CHAPTER CROSSREF EDITION

1 MANUAL my-citat… Vienna…

2 ARTICLE refmanag…

3 MANUAL jqr

# … with 19 more variables: EDITOR , HOWPUBLISHED ,

# INSTITUTION , JOURNAL , KEY , MONTH , NOTE ,

# NUMBER , ORGANIZATION , PAGES , PUBLISHER ,

# SCHOOL , SERIES , TITLE , TYPE , VOLUME ,

# YEAR , URL , DOI

bib2df::df2bib(df)

@Manual{my-citation-key-for-r,

Address = {Vienna, Austria},

Author = {R Core Team},

Organization = {R Foundation for Statistical Computing},

Title = {R: A Language and Environment for Statistical Computing},

Year = {2020},

Url = {<https://www.R-project.org/>}

}

@Article{refmanager,

Author = {Mathew William McLean},

Journal = {The Journal of Open Source Software},

Title = {RefManageR: Import and Manage BibTeX and BibLaTeX References in R},

Year = {2017},

Doi = {10.21105/joss.00338}

}

@Manual{jqr,

Author = {Rich FitzJohn and Jeroen Ooms and Scott Chamberlain and {Stefan Milton Bache},

Note = {R package version 1.1.0},

Title = {jqr: Client for 'jq', a 'JSON' Processor},

Year = {2018},

Url = {<https://CRAN.R-project.org/package=jqr>}

}

Note that RefManageR has a similar export function.

library("magrittr")

df2 <- "refs.bib" %>%

RefManageR::ReadBib() %>%

as.data.frame()

df2

bibtype

my-citation-key-for-r Manual

refmanager Article

jqr Manual

title

my-citation-key-for-r R: A Language and Environment for Statistical Computing

refmanager RefManageR: Import and Manage BibTeX and BibLaTeX References in R

jqr jqr: Client for 'jq', a 'JSON' Processor

author

my-citation-key-for-r {R Core Team}

refmanager Mathew William McLean

jqr Rich FitzJohn and Jeroen Ooms and Scott Chamberlain and {Stefan Milton Bache}

organization address

my-citation-key-for-r R Foundation for Statistical Computing Vienna, Austria

refmanager

jqr

year url

my-citation-key-for-r 2020 <https://www.R-project.org/>

refmanager 2017

jqr 2018 <https://CRAN.R-project.org/package=jqr>

journal doi

my-citation-key-for-r

refmanager The Journal of Open Source Software 10.21105/joss.00338

jqr

note

my-citation-key-for-r

refmanager

jqr R package version 1.1.0

A difference in the formats is for instance the way authors are parsed.

df$AUTHOR

[[1]]

[1] "R Core Team"

[[2]]

[1] "Mathew William McLean"

[[3]]

[1] "Rich FitzJohn" "Jeroen Ooms" "Scott Chamberlain"

[4] "{Stefan Milton Bache"

df2$author

[1] "{R Core Team}"

[2] "Mathew William McLean"

[3] "Rich FitzJohn and Jeroen Ooms and Scott Chamberlain and {Stefan Milton Bache}"

bib2df even supports separating names,

bib2df::bib2df("refs.bib", separate\_names = TRUE)$AUTHOR

[[1]]

salutation first\_name middle\_name last\_name suffix full\_name

1 R Core Team R Core Team

[[2]]

salutation first\_name middle\_name last\_name suffix full\_name

1 Mathew William McLean Mathew William McLean

[[3]]

salutation first\_name middle\_name last\_name suffix full\_name

1 Rich FitzJohn Rich FitzJohn

2 Jeroen Ooms Jeroen Ooms

3 Scott Chamberlain Scott Chamberlain

4 {Stefan Milton Bache {Stefan Milton Bache

bib2df helps doing fun or serious analyses of reference databases.

handlr is less mature but not less useful.  
It’s *a tool for converting among citation formats*,.

citation <- handlr::HandlrClient$new(x = "refs.bib")

citation

doi:

ext: bib

format (guessed): bibtex

path: refs.bib

string (abbrev.): none

citation$write(format = "ris")

[1] "TY - GEN\r\nT1 - R: A Language and Environment for Statistical Computing\r\nAU - R Core Team\r\nUR - [https://www.R-project.org/\r\nER](https://www.R-project.org/%5Cr%5CnER) - "

[2] "TY - JOUR\r\nT1 - RefManageR: Import and Manage BibTeX and BibLaTeX References in R\r\nAU - McLeanMathew William\r\nDO - 10.21105/joss.00338\r\nER - "

[3] "TY - GEN\r\nT1 - jqr: Client for 'jq', a 'JSON' Processor\r\nAU - FitzJohnRich\r\nAU - OomsJeroen\r\nAU - ChamberlainScott\r\nAU - Stefan Milton Bache\r\nUR - [https://CRAN.R-project.org/package=jqr\r\nER](https://CRAN.R-project.org/package=jqr%5Cr%5CnER) - "

At the moment, there are supported readers for CiteProc, RIS, BibTeX, CodeMeta and supported writers for CiteProc, RIS, BibTeX, RDF/XML, CodeMeta. Quite an arsenal for your bibliography conversion needs!