# The Emacs Org-mode: Reproducible Research and Beyond

**Andreas Leha**[1,⋆], **Tim Beißbarth**[1]

1. Department of Medical Statistics, University Medical Center Göttingen
⋆Contact author: andreas.leha@med.uni-goettingen.de

**Keywords:** Literate Programming, Reproducible Research, emacs, Org-mode

Reproducible research and literate programming are widespread techniques in the R community using the power of the well established Sweave package [Leisch, 2002] and other approaches (e.g. R2HTML [Lecoutre, 2003], odfWeave [Kuhn et al., 2010], via docutils [Dasgupta, 2010]). The main advantage of all literate programming efforts is, that (R-)code and results, mainly tables and plots, are kept together. This ensures that the reported results originate from the current version of the code, making the report reproducible and adding other benefits, like simplifying version control for results and code in conjunction.

Org-mode (http://orgmode.org/) can be used to achieve similar results as Sweave. But while in contrast to Sweave Org-mode is bound to the editor emacs (although there are different ports to vi(m)), it is a far more flexible approach in other aspects:

The most prominent feature is the variability of output formats: the output can be chosen from LaTeX (PDF), HTML, and soon also ODF.

The language that is woven into the text is not limited to R, but includes Perl, Python, Octave, shell, Matlab, Lisp, SQL, and many more. This is a key feature, as quite often data analysis requires the use of different tools on the same data: For example, Perl is frequently used during the data pre-processing step. Likewise, external programs are often integrated for specialized tasks (e.g. short read alignment) and might best be called from command line. Source code blocks in different languages can easily interact within Org-mode by passing the output from one block as input to another block. Results from a code block can be cached to be re-calculated only when the code changed, avoiding repeated execution of lengthy calculations.

It is even possible to do metaprogramming in Org-mode, meaning that one block of source code generates source code again, which will be handled correctly by Org-mode, too.

And finally, Org-mode is much more than a tool for literate programming. Being initially developed for outlining, (TODO) lists, and project and task management, all this functionality is also available to the R programmer.

We present workflows of how to perform data analyses, create reports, and publish results on your website from within one tool.

## References

Abhijit Dasgupta. Flexible report generation and literate programming using r and python's docutils module. In *useR!*, 2010. URL http://user2010.org/slides/Dasgupta.pdf.

Max Kuhn, Steve Weston, Nathan Coulter, Patrick Lenon, and Zekai Otles. *odfWeave: Sweave processing of Open Document Format (ODF) files*, 2010. URL http://CRAN.R-project.org/package=odfWeave. R package version 0.7.17.

Eric Lecoutre. The R2HTML package. *R News*, 3(3):33–36, December 2003. URL http://cran.r-project.org/doc/Rnews/Rnews_2003-3.pdf.

Friedrich Leisch. Sweave: Dynamic generation of statistical reports using literate data analysis. In Wolfgang Härdle and Bernd Rönz, editors, *Compstat 2002 — Proceedings in Computational Statistics*, pages 575–580. Physica Verlag, Heidelberg, 2002. URL http://www.stat.uni-muenchen.de/~leisch/Sweave. ISBN 3-7908-1517-9.