The package RJournal1c

Kurt Hornik and Friedrich Leisch
(with subsequent edits by the editorial board)

January 11, 2013

1 Introduction

The LaTeX 2_ε package RJournal1c provides commands for formatting The R Journal.

2 Documentation

2.1 Marking Words and Phrases

The RJournal package provides roughly the same commands for marking words and phrases as does Texinfo (but note that the LaTeX special characters still need special treatment). The standard commands are as follows:

`\code`{*sample-code*} Indicate text that is a literal example of a piece of a program.

`\samp`{*text*} Indicate text that is a literal example of a sequence of characters.

`\file`{*file-name*} Indicate the name of a file.

`\dfn`{*term*} Indicate the introductory or defining use of a term.

The following commands are defined for completeness, but not recommended

`\kbd`{*keyboard-characters*} Indicate keyboard input.

`\key`{*key-name*} Indicate the conventional name for a key on a keyboard.

`\var`{*metasyntactic-variable*} Indicate a metasyntactic variable.

`\env`{*environment-variable*} Indicate an environment variable.

`\command`{*command-name*} Indicate a command name (such as '`ls`').

`\option`{*option-name*} Indicate a command line option.

`\acronym`{*acronym*} Use for abbreviattions written in all capital letters, such as '`NASA`'.

If this sounds rather confusing, please see the Texinfo documentation for more details.

`\strong` There is also a `\strong` command for emphasizing text more strongly than with `\emph`. For example, `\strong{Note:}` gives **Note:**.

`\pkg` `\CRANpkg` To indicate R packages use `\pkg` or `\CRANpkg` (alias `\cpkg`): the latter adds a hyperlink to the package's CRAN page and is recommended at least for the first mention of a CRAN package, particularly when a full citation is not warranted.

`\ctv` Similarly, task views may be indicated by `\ctv`.

2.2 Quotations and Examples

In addition to the standard LaTeX for quotations and examples (such as `quote`, `quotation`, `flushleft`, `center` and `flushright`), the RJournal1c package provides the following environments.

`example` Illustrate code, commands, and the like. The text is printed in a fixed-width font, and indented but not filled.

`smallexample` Similar to `example`, except that text is typeset in a smaller font.

These are patterned after the Texinfo environments with the same names. In particular, `{`, `}`, `\` retain their "usual" meanings and are not treated verbatim, which is not optimal for displaying R code or output. Hence, we also provide a `smallverbatim` environment which works like `verbatim` but uses a smaller font for typesetting.

3 The Code

3.1 The Batch File

First comes the code for creating the batch file 'RJournal1c.ins' which in turn can be used for producing the package and driver files.

```
⟨*install⟩
\begin{filecontents}{\filename.ins}
% Simply TeX or LaTeX this file to extract various files from the source
% file 'RJournal.dtx'.
  \def\filedate{2001/01/05} \def\batchfile{RJournal.ins} \input
  docstrip.tex \preamble
\endpreamble
\generateFile{RJournal.drv}{t}{\from{RJournal.dtx}{driver}}
\generateFile{RJournal.sty}{t}{\from{RJournal.dtx}{package}}
\Msg{*****************************************************************}
\Msg{* For documentation, run LaTeX on RJournal.dtx or RJournal.drv. *}
\Msg{*****************************************************************}
\end{filecontents}
⟨/install⟩
```

3.2 The Driver

Next comes the documentation driver file for TEX, i.e., the file that will produce the documentation you are currently reading. It will be extracted from this file by the `docstrip` program. Since it is the first code in the file one can alternatively process this file directly with LATEX 2_ε to obtain the documentation.

```
⟨*driver⟩
\documentclass[fleqn]{ltxdoc}
\usepackage[driver]{\filename}
\renewcommand{\pkg}[1]{\textsf{#1}}
\begin{document}
  \DocInput{\filename.dtx}
\end{document}
⟨/driver⟩
```

3.3 The Code

Now comes the code for the package.

If the current format is not LATEX 2_ε, we abort immediately. Otherwise, we provide ourselves and show the current version of the package on the screen and in the transscript file.

```
⟨*package⟩
\NeedsTeXFormat{LaTeX2e}[1995/12/01]
\ProvidesPackage{\filename}[\filedate\space\fileversion\space
                        RJournal package]
\typeout{Package: `\filename\space\fileversion \@spaces <\filedate>'}
\typeout{English documentation as of <\docdate>}
```

Then we load the hyperref package and format the back-links that will appear in the bibliography

```
\RequirePackage[pagebackref]{hyperref}
\renewcommand{\backref}[1]{[p#1]}
```

Next, we set up a more or less trivial option handler. We use option 'driver' for conditionalizing package code we do not want executed when typesetting the driver file.

```
\RequirePackage{ifthen}
\newboolean{RJ@driver}
\DeclareOption{driver}{\setboolean{RJ@driver}{true}}
\DeclareOption*{\PackageWarning{\filename}{Unknown option
    `\CurrentOption'}}
\ProcessOptions\relax
```

Now comes the real code.

```
\ifthenelse{\boolean{RJ@driver}}{}{
```

First we load some utility packages.

```
\RequirePackage{tikz}
\RequirePackage{graphicx,color,fancyhdr}
\RequirePackage{amsmath}
```

```
\RequirePackage{placeins}
\IfFileExists{upquote.sty}{\RequirePackage{upquote}}{}
```

3.3.1 Basic Structure

Issues of of *The R Journal* are created from the standard LaTeX document class report. Individual articles correspond to chapters, and are contained in `article` environments. This makes it easy to have figures counted within articles and hence hyperlinked correctly.

Basic front matter information about the issue: volume, number, and date.

```
\newcommand{\volume}[1]{\def\RJ@volume{#1}}
\newcommand{\volnumber}[1]{\def\RJ@number{#1}}
\renewcommand{\date}[1]{\def\RJ@date{#1}}
```

We do not want numbered sections.

```
\setcounter{secnumdepth}{-1}
```

`\sectionhead` `\author` `\title` `\subtitle`

An article has an author, a title, and optionally a subtitle. We use the obvious commands for specifying these. Articles will be put in certain journal sections, named by `\sectionhead`.

```
\newcommand{\sectionhead}[1]{\def\RJ@sectionhead{#1}}
\renewcommand{\author}[1]{\def\RJ@author{#1}}
\renewcommand{\title}[1]{\def\RJ@title{#1}}
\newcommand{\subtitle}[1]{\def\RJ@subtitle{#1}}
```

Note that we put the title info in the TOC and the bookmarks when creating PDF. Thus titles should really only contain text.

`article`

Environment `article` clears the article header information at its beginning. We use `\FloatBarrier` from the placeins package to keep floats within the article.

```
\newenvironment{article}{\author{}\title{}\subtitle{}\FloatBarrier}{\FloatBarrier}
```

`\maketitle`

The real work is done by a redefined version of `\maketitle`. Note that even though we do not want chapters (articles) numbered, we need to increment the chapter counter, so that figures get correct labelling.

```
\renewcommand{\maketitle}{
    \chapter{\RJ@title}\refstepcounter{chapter}
    \ifx\empty\RJ@subtitle\else\noindent\textbf{\RJ@subtitle}
    \par\nobreak\addvspace{\baselineskip}\fi
    \ifx\empty\RJ@author\else\noindent\textit{\RJ@author}
    \par\nobreak\addvspace{\baselineskip}\fi
    \@afterindentfalse\@nobreaktrue\@afterheading}
```

Now for some ugly redefinitions. We do not want articles to start a new page. (Actually, we do, but this is handled via explicit

The name@of@eq is a hack to get hyperlinks to equations to work within each article, even though there may be multiple eq.(1)

```
\renewcommand\chapter{\secdef\RJ@chapter\@schapter}
\providecommand{\nohyphens}{%
  \hyphenpenalty=10000\exhyphenpenalty=10000\relax}
\newcommand{\RJ@chapter}{%
  \edef\name@of@eq{equation.\@arabic{\c@chapter}}%
  \renewcommand{\@seccntformat}[1]{}%
  \@startsection{chapter}{0}{0mm}{%
    -2\baselineskip \@plus -\baselineskip \@minus -.2ex}{\p@}{%
    \phantomsection\normalfont\Huge\bfseries\raggedright}}
```

TOC entries for articles (chapters) should really look like sections.

```
\renewcommand*\l@chapter{\@dottedtocline{0}{0pt}{1em}}
```

Book reviews should appear as sections in the text and in the pdf bookmarks, however we wish them to appear as chapters in the TOC. Thus we define an alternative to `\maketitle` for reviews.

Refereed articles should have an abstract, so we redefine `\abstract` to give the desired style

```
\renewcommand{\abstract}[1]{\begin{center}\begin{minipage}{0.92\linewidth}
    \textbf{Abstract } #1 \end{minipage}\end{center} \subsubsection{} \vspace{-12pt}}
```

We want bibliographies as starred sections within articles. As the standard `thebibliography` environment uses `\chapter*`, we simply redefine the latter according to our needs.

```
\def\@schapter#1{\section*#1}
```

Equations, figures and tables are counted within articles, but we do not show the article number. For equations it becomes a bit messy to avoid having hyperref getting it wrong.

```
\numberwithin{equation}{chapter}
\renewcommand{\theequation}{\@arabic\c@equation}
\renewcommand{\thefigure}{\@arabic\c@figure}
\renewcommand{\thetable}{\@arabic\c@table}
```

`\tableofcontents` Need to provide our own version of `\tableofcontents`. We use the tikz package to get the rounded rectangle. Notice that `\section*` is really the same as `\chapter*`.

```
\renewcommand{\contentsname}{Contents}
\renewcommand\tableofcontents{%
\begin{center}
\vspace{1cm}
\begin{tikzpicture}
\node[right,text width=15.5cm, draw=black, rounded corners=20pt,
fill=white,inner sep=2em, very thick]
{
  \begin{minipage}[t][16.5cm][c]{1.0\linewidth}
  \section*{\contentsname}
 {
  \@starttoc{toc}
```

```
 }
  \end{minipage}
};

\end{tikzpicture}
\end{center}}
```

\titlepage The title page of each issue features logo et al at the top and the TOC. We start with the top, defining metadata for the pdf first.

```
\renewcommand{\titlepage}{%
  \hypersetup{
    pdftitle={The R Journal Volume \RJ@volume/\RJ@number, \RJ@date},%
    pdfauthor={R Foundation for Statistical Computing},%
  }
  \noindent
  \begin{center}
    \fontsize{60pt}{60pt}\selectfont
    The \raisebox{-8pt}{\includegraphics[height=77pt]{Rlogo-4}}\hspace{10pt}
    Journal
  \end{center}
  {\large\hfill Volume \RJ@volume/\RJ@number, \RJ@date\qquad}
  \\[-.5\baselineskip]
  \rule{\textwidth}{1pt}
  \begin{center}
    {\Large A peer-reviewed, open-access publication of the R Foundation\\
      for Statistical Computing}
  \end{center}
```

Now set up the header and footer information for the rest of the document.

```
  \fancyhf{}
  \fancyhead[LO,RE]{\textsc{\RJ@sectionhead}}
  \fancyhead[RO,LE]{\thepage}
  \fancyfoot[L]{The R Journal Vol. \RJ@volume/\RJ@number, \RJ@date}
  \fancyfoot[R]{ISSN 2073-4859}
  \thispagestyle{empty}
```

And finally, put in the TOC box. Note the way `tocdepth` is adjusted before and after producing the TOC: thus, we can ensure that only articles show up in the printed TOC, but that in the PDF version, bookmarks are created for sections and subsections as well (provided that the non-starred forms are used).

```
\setcounter{tocdepth}{0}
\tableofcontents
\setcounter{tocdepth}{2}
\clearpage
}
```

3.3.2 Layout, Fonts and Color

Layout. We set the basic layout parameters in a way that printouts should be fine for both A4 and Letter paper.

```
\setlength{\textheight}{250mm}
\setlength{\topmargin}{-10mm}
\setlength{\textwidth}{17cm}
\setlength{\oddsidemargin}{-6mm}
\setlength{\evensidemargin}{-6mm}
```

Fonts. We use the following fonts (all with T1 encoding):

rm	palatino
tt	courier (condensed)
sf	almost european
sf	helvetica
math	palatino

```
\RequirePackage{pslatex}
\RequirePackage{palatino,mathpazo}
\RequirePackage[T1]{fontenc}
```

Colors. We define a dark blue color for all links.

```
\definecolor{link}{rgb}{0,0,0.3}
\hypersetup{
  colorlinks,%
  citecolor=link,%
  filecolor=link,%
  linkcolor=link,%
  urlcolor=link
}
```

3.3.3 Miscellania

```
\newcommand{\R}{R}
\newcommand{\address}[1]{\addvspace{\baselineskip}\noindent\emph{#1}}
\newcommand{\email}[1]{\href{mailto:#1}{\normalfont\texttt{#1}}}
```

`boxedverbatim` Used for creating a boxed (small) verbatim environment. The code is taken from package moreverb. Note that we need to use `verbatim` rather than `alltt`.

```
\RequirePackage{verbatim}
\def\boxedverbatim{%
  \def\verbatim@processline{%
    {\setbox0=\hbox{\the\verbatim@line}%
    \hsize=\wd0 \the\verbatim@line\par}}%
  \@minipagetrue
  \@tempswatrue
  \setbox0=\vbox
  \bgroup\small\verbatim
}
\def\endboxedverbatim{%
  \endverbatim
  \unskip\setbox0=\lastbox
```

```
  \egroup
  \fbox{\box0}
}
```

Finally, we turn on fancy page style.

```
\pagestyle{fancy}
} % \ifthenelse{\boolean{RJ@driver}}
```

3.3.4 Marking Words and Phrases

Simple font selection is not good enough. For example, `\texttt{--}` gives '--', i.e., an endash in typewriter font. Hence, we need to turn off ligatures, which currently only happens for commands `\code` and `\samp` and the ones derived from them. Hyphenation is another issue; it should really be turned off inside `\samp`. And most importantly, LaTeX special characters are a nightmare. E.g., one needs `\~{}` to produce a tilde in a file name marked by `\file`. Perhaps a few years ago, most users would have agreed that this may be unfortunate but should not be changed to ensure consistency. But with the advent of the WWW and the need for getting '~' and '#' into URLs, commands which only treat the escape and grouping characters specially have gained acceptance (in fact, this is also what `alltt` does, and hence environments based on it such as our `smallexample`). Hence, in the long run we should implement the same for `\code`, `\kbd`, `\samp`, `\var`, and `\file`. (The other Texinfo-style commands do not need this.)

```
\DeclareRobustCommand\code{\bgroup\@noligs\@codex}
\def\@codex#1{\texorpdfstring%
{{\normalfont\ttfamily\hyphenchar\font=-1 #1}}%
{#1}\egroup}
\newcommand{\kbd}[1]{{\normalfont\texttt{#1}}}
\newcommand{\key}[1]{{\normalfont\texttt{\uppercase{#1}}}}
\DeclareRobustCommand\samp{`\bgroup\@noligs\@sampx}
\def\@sampx#1{{\normalfont\texttt{#1}}\egroup'}
\newcommand{\var}[1]{{\normalfont\textsl{#1}}}
\let\env=\code
\newcommand{\file}[1]{{`\normalfont\textsf{#1}'}}
\let\command=\code
\let\option=\samp
\newcommand{\dfn}[1]{{\normalfont\textsl{#1}}}
% \acronym is effectively disabled since not used consistently
\newcommand{\acronym}[1]{#1}
\newcommand{\strong}[1]{\texorpdfstring%
{{\normalfont\fontseries{b}\selectfont #1}}%
{#1}}
\let\pkg=\strong
\newcommand{\CRANpkg}[1]{\href{http://cran.r-project.org/package=#1}{\pkg{#1}}}%
\let\cpkg=\CRANpkg
\newcommand{\ctv}[1]{\href{http://CRAN.R-project.org/view=#1}{\emph{#1}}}
```

3.3.5 Quotations and Examples

```
\RequirePackage{alltt}
\newenvironment{example}{\begin{alltt}}{\end{alltt}}
\newenvironment{smallexample}{\begin{alltt}\small}{\end{alltt}}
\newenvironment{display}{\list{}{}\item\relax}{\endlist}
\newenvironment{smallverbatim}{\small\verbatim}{\endverbatim}
```

Support for output from Sweave, and generic session style code These used to have fontshape=sl for Sinput/Scode/Sin, but pslatex won't use a condensed font in that case.

```
\RequirePackage{fancyvrb}
\DefineVerbatimEnvironment{Sinput}{Verbatim}{fontsize=\small}
\DefineVerbatimEnvironment{Soutput}{Verbatim}{fontsize=\small}
\DefineVerbatimEnvironment{Scode}{Verbatim}{fontsize=\small}
\DefineVerbatimEnvironment{Sin}{Verbatim}{fontsize=\small}
\DefineVerbatimEnvironment{Sout}{Verbatim}{fontsize=\small}
\newenvironment{Schunk}{}{}
```

3.3.6 Mathematics

`\operatorname` The implementation of `\operatorname` is similar to the mechanism LaTeX 2_ε uses for functions like sin and cos, and simpler than the one of $\mathcal{AMS}$-LaTeX. We use `\providecommand` for the definition in order to keep the one of the amstex if this package has already been loaded.

```
\providecommand{\operatorname}[1]{%
  \mathop{\operator@font#1}\nolimits}
```

`\P` `\E` `\VAR` `\COV` `\COR` Next, we provide commands for probability, expectation, variance, covariance and correlation which are obviously useful in probability theory and statistics. (Of course, originally `\P` gives ¶.)

```
\renewcommand{\P}{%
  \mathop{\operator@font I\hspace{-1.5pt}P\hspace{.13pt}}}
\newcommand{\E}{%
  \mathop{\operator@font I\hspace{-1.5pt}E\hspace{.13pt}}}
\newcommand{\VAR}{\operatorname{var}}
\newcommand{\COV}{\operatorname{cov}}
\newcommand{\COR}{\operatorname{cor}}
```

Finally, we load package amsfonts so that `\mathbb` is available for producing the symbols for positive integers etc.

```
\RequirePackage{amsfonts}
```

This ends the implementation of the RJournal1c package.

⟨/package⟩