

The ragged2e-package*

Martin Schröder

<https://gitlab.com/TeXhackse/ragged2e>[†]

2021/02/05

Abstract

This package provides new commands and environments for setting ragged text which are easy to configure to allow hyphenation. An earlier attempt to do this was the style `raggedright` [9] by the same author.

Contents

1	The problem	2
2	Old “solutions”	2
2.1	<code>L^AT_EX</code>	2
2.2	<code>plain</code>	2
3	Our solution	3
3.1	The macros	3
3.2	The parameters	3
3.3	The environments	5
4	Options	5
5	Required packages	6
6	The implementation	6
6.1	Initial Code	6
6.2	Declaration of options	7
6.2.1	<code>originalcommands</code> option	7
6.2.2	<code>originalparameters</code> option	7
6.2.3	<code>raggedrightboxes</code> option	7
6.2.4	<code>footnotes</code> option	8
6.2.5	<code>document</code> option	8
6.2.6	Other options	9
6.3	Executing options	9
6.4	Loading packages	9
6.5	Allocations	10

*The version number of this file is v3.0, revision #608e643, last revised 2021-02-05 20:21:49 +0100.

[†]maintained by Marei Peischl

6.6 Initializations 10
 6.7 Distinguishing between monospaced and proportional fonts 11
 6.8 The commands 12
 6.9 The environments 14
 6.10 Overloading the L^AT_EX-commands 14

7 Acknowledgements 15

1 The problem

L^AT_EX has three commands (`\centering`, `\raggedleft`, and `\raggedright`) and three environments (`center`, `flushleft`, and `flushright`) to typeset ragged text. The environments are based upon the commands (`center` uses `\centering`, `flushleft` `\raggedright`, and `flushright` `\raggedleft`).

These commands have, however, one serious flaw: they render hyphenation almost impossible, and thus the text looks *too* ragged, as the following example shows:

<code>\raggedright:</code> “The L ^A T _E X document preparation system is a special version of Donald Knuth’s T _E X program. T _E X is a sophisticated program designed to produce high-quality typesetting, especially for mathematical text.” [5, p. xiii]	<code>\RaggedRight:</code> “The L ^A T _E X document preparation system is a special version of Donald Knuth’s T _E X program. T _E X is a sophisticated program designed to produce high-quality typesetting, especially for mathematical text.” [5, p. xiii]
---	---

2 Old “solutions”

2.1 L^AT_EX

L^AT_EX defines e. g. `\raggedright` as follows:

```
1 (*latex)
```

```
2 \def\raggedright{%
3   \let\=\@centercr
4   \@rightskip\@flushglue
5   \rightskip\@rightskip
6   \leftskip\z@
7   \parindent\z@}
```

Initially, `\@flushglue` is defined as

```
8 \@flushglue = Opt plus 1fil
9 \</latex>
```

Thus the `\rightskip` is set to `Opt plus 1fil`. Knuth, however warns [4, p. 101]:

“For example, a person can set `\rightskip=Opt plus 1fil`, and every line will be filled with space to the right. But this isn’t a particularly good way to make ragged-right margins, because the infinite stretchability will assign zero badness to lines that are very short. To do a decent job of ragged-right setting, the trick is to set `\rightskip` so that it will stretch enough to make line breaks possible, yet not too much, because short lines should be considered bad. Furthermore the spaces between words should be fixed so that they do not stretch or shrink.”

2.2 plain

plain T_EX defines an special version of `\raggedright`, which operates the way Knuth describes

```

10 it; but which can not be used with
    LATEX, because LATEX redefines
    \raggedright.
11 (*plain)
12 \def\raggedright{%
13   \rightskip\z@ plus2em
14   \spaceskip.3333em
15   \xspaceskip.5em\relax}
    plain provides also a version of
    \raggedright for typewriter fonts
16 \def\tttraggedright{%
17   \tt
18   \rightskip\z@ plus2em\relax}
19 \</plain>

```

3 Our solution

Since the `plain` solution can not be used with L^AT_EX, we have to redefine it and make it possible to configure it for personal preferences.

3.1 The macros

`\Centering`, `\RaggedLeft`, and `\RaggedRight` can be used in the same way as `\centering`, `\raggedleft`, and `\raggedright`: Just type the command, and after that the whole text will be set centered, ragged-left or ragged-right.

For example, we switched on `\RaggedRight` on the top of this text, and consequently this text was set ragged-right.*

`\justifying` switches back to justified text after ragged text has been switched on.

The new commands `\Centering`, `\RaggedLeft`, and `\RaggedRight` are fully compatible with their counterparts in L^AT_EX, but implement the `plain` solution and can be easily configured using the following parameters:

3.2 The parameters

Command	Uses
<code>\Centering</code>	<code>\CenteringLeftskip</code> , <code>\CenteringRightskip</code> , <code>\CenteringParfillskip</code> , <code>\CenteringParindent</code>
<code>\RaggedLeft</code>	<code>\RaggedLeftLeftskip</code> , <code>\RaggedLeftRightskip</code> , <code>\RaggedLeftParfillskip</code> , <code>\RaggedLeftParindent</code>
<code>\RaggedRight</code>	<code>\RaggedRightLeftskip</code> , <code>\RaggedRightRightskip</code> , <code>\RaggedRightParfillskip</code> , <code>\RaggedRightParindent</code>
<code>\justifying</code>	<code>\JustifyingParfillskip</code> , <code>\JustifyingParindent</code>

All Parameters can be set with `\setlength`, e. g.

```
\setlength{\RaggedRightRightskip}{0pt plus 1em}
```

*For this documentation we also set `\RaggedRightRightskip` higher than usual (`0pt plus 4em` instead of `0pt plus 2em`) because of all the long command names which make linebreaking difficult.

sets `\RaggedRightRightskip` to `0pt plus 1em`.

`\CenteringLeftskip` `\RaggedLeftLeftskip` `\RaggedRightLeftskip` These are the `\leftskips` inserted by `\Centering`, `\RaggedLeft`, and `\RaggedRight`.

“`\leftskip` (glue at left of justified lines)” [4, p. 274]

`\leftskip` must be set to a finite value, to make hyphenation possible. Setting it to infinite values like `0pt plus 1fil` makes hyphenation almost impossible.

`\CenteringRightskip` `\RaggedLeftRightskip` `\RaggedRightRightskip` These are the `\rightskips` inserted by `\Centering`, `\RaggedLeft`, and `\RaggedRight`.

“`\rightskip` (glue at right of justified lines)” [4, p. 274]

`\rightskip` must be set to a finite value, to make hyphenation possible. Setting it to infinite values like `0pt plus 1fil` makes hyphenation almost impossible.

`\CenteringParfillskip` `\RaggedLeftParfillskip` `\RaggedRightParfillskip` `\JustifyingParfillskip` These are the `\parfillskips` inserted by `\Centering`, `\RaggedLeft`, `\RaggedRight`, and `\justifying`.

“`\parfillskip` (additional `\rightskip` at end of paragraphs)” [4, p. 274]

The normal setting for `\parfillskip` is `0pt plus 1fil`; the parameters are provided for testing combinations of `\{left|right\}skip` and `\parfillskip`.

`\CenteringParindent` `\RaggedLeftParindent` `\RaggedRightParindent` `\JustifyingParindent` These are the `\parindents` used by `\Centering`, `\RaggedLeft`, `\RaggedRight`, and `\justifying`.

“`\parindent` (width of `\indent`)” [4, p. 274]

`\parindent` is the indent of the first line of a paragraph and should be set to `0pt`, since indented lines in ragged text do not look good.

The parameters have the following initial setting:

Parameter	L ^A T _E X setting	ragged2e setting [†]
<code>\CenteringLeftskip</code>	Opt plus 1fil	Opt plus 2em
<code>\RaggedLeftLeftskip</code>	Opt plus 1fil	Opt plus 2em
<code>\RaggedRightLeftskip</code>	Opt plus Opt minus Opt	Opt plus Opt minus Opt
<code>\CenteringRightskip</code>	Opt plus 1fil	Opt plus 2em
<code>\RaggedLeftRightskip</code>	Opt plus Opt minus Opt	Opt plus Opt minus Opt
<code>\RaggedRightRightskip</code>	Opt plus 1fil	Opt plus 2em
<code>\CenteringParfillskip</code>	Opt plus Opt minus Opt	Opt plus Opt minus Opt
<code>\RaggedLeftParfillskip</code>	Opt plus Opt minus Opt	Opt plus Opt minus Opt
<code>\RaggedRightParfillskip</code>	Opt plus 1fil	Opt plus 1fil
<code>\CenteringParindent</code>	Opt	Opt
<code>\RaggedLeftParindent</code>	Opt	Opt
<code>\RaggedRightParindent</code>	Opt	Opt
<code>\JustifyingParfillskip</code>		Opt plus 1fil
<code>\JustifyingParindent</code>		<code>\parindent</code>

3.3 The environments

- Center** Center is fully compatible with `center`, but uses `\Centering` instead of `\centering`.
- FlushLeft** `FlushLeft` is fully compatible with `flushleft`, but uses `\RaggedRight` instead of `\raggedright`.
- FlushRight** `FlushRight` is fully compatible with `flushright`, but uses `\RaggedLeft` instead of `\raggedleft`.
- justify** `justify` is like the other environments but uses `\justifying`.

New feature
2003/01/04

E. g. `FlushLeft` can be used in the same way as `flushleft`:

```
\begin{FlushLeft}
<text, which is set ragged-right>
\end{FlushLeft}
```

4 Options

This package has the following options:

`originalcommands` The L^AT_EX-commands `\centering`, `\raggedleft`, and `\raggedright` and the L^AT_EX-environments `center`, `flushleft`, and `flushright` remain unchanged.

It is the default.

[†]For proportional and monospaced fonts.

newcommands The L^AT_EX-commands `\centering`, `\raggedleft`, and `\raggedright` and the L^AT_EX-environments `center`, `flushleft`, and `flushright` are set equal to their counterparts defined by `ragged2e`. Thus `\raggedright` invokes `\RaggedRight`. The original commands can be accessed under the Names `\LaTeX \langle original name \rangle` , e. g. `\LaTeXraggedright`.

originalparameters The parameters used by the commands implemented by `ragged2e` are initialized with the default settings used by L^AT_EX.

newparameters The parameters used by the commands implemented by `ragged2e` are initialized with the default settings defined by `ragged2e`.

It is the default.

New feature
2003/01/18

raggedrightboxes All `\parboxes`, `minipages`, `\marginpars` and p-columns of `tabulars` and `arrays` are automatically set using `\RaggedRight`.

New feature
2003/01/18

footnotes This options sets all footnotes ragged-right by loading the `footmisc` [2] package with the `ragged` option.

New feature
2003/01/18

document This options sets the complete document ragged-right by executing a `\RaggedRight` at `\begin{document}` and the `raggedrightboxes` and the `footnotes` options.

All other options are passed to the `footmisc` package if the `footnotes` option is selected.

5 Required packages

This package requires the following packages:

everyysel [8] (only if format older than 2021/01/05) It has been used to distinguish between monospaced and proportional fonts as long as the L^AT_EX kernel did not provide the functionality with `lthooks` [6]. Formats newer than 2021/01/05 do no longer depend on `everyysel`.

footmisc [2] It is used by the `footnotes` and the `document` options; at least version 5.00 (2002/08/28) is needed.

6 The implementation

²⁰ `(*package)`

6.1 Initial Code

`\if@raggedtwoe@originalcommands`

`\if@raggedtwoe@originalcommands` is used to flag the use of the original-commands or `newcommands` option.

²¹ `\newif\if@raggedtwoe@originalcommands`

```

\if@raggedtwoe@originalparameters \if@raggedtwoe@originalparameters is used to flag the use of the
originalparameters or newparameters option.
22 \newif\if@raggedtwoe@originalparameters

\if@raggedtwoe@footmisc \if@raggedtwoe@footmisc is used to flag the use of the footnotes option.
23 \newif\if@raggedtwoe@footmisc

```

6.2 Declaration of options

6.2.1 originalcommands option

The `originalcommands` and `newcommands` options control the meaning of the \LaTeX -commands for ragged text: If `newcommands` is used the \LaTeX -commands are set equal to the commands defined by `ragged2e`.

```

24 \DeclareOption{OriginalCommands}{\@raggedtwoe@originalcommandstrue}
25 \DeclareOption{originalcommands}{\@raggedtwoe@originalcommandstrue}
26 \DeclareOption{NewCommands}{\@raggedtwoe@originalcommandsfalse}
27 \DeclareOption{newcommands}{\@raggedtwoe@originalcommandsfalse}

```

6.2.2 originalparameters option

The `originalparameters` and `newparameters` options control the defaults for the parameters used by the commands implemented by `ragged2e`: If `newparameters` is used the parameters are set to the values defined by `ragged2e`.

```

28 \DeclareOption{OriginalParameters}{\@raggedtwoe@originalparameterstrue}
29 \DeclareOption{originalparameters}{\@raggedtwoe@originalparameterstrue}
30 \DeclareOption{NewParameters}{\@raggedtwoe@originalparametersfalse}
31 \DeclareOption{newparameters}{\@raggedtwoe@originalparametersfalse}

```

6.2.3 raggedrightboxes option

The option `raggedrightboxes` sets all `\parboxes`, `minipages`, `\marginpars` and `p-columns` of `tabulars` and `arrays` using `\RaggedRight`. This is done by redefining `\@arrayparboxrestore`.

```

\@raggedtwoe@raggedrightboxes@opt \@raggedtwoe@raggedrightboxes@opt is the code executed via
\DeclareOption.

```

```

32 \newcommand*{\@raggedtwoe@raggedrightboxes@opt}{

```

First we check if `\@arrayparboxrestore` is unchanged.

```

33 \CheckCommand*{\@arrayparboxrestore}{\%
34 \let\if@nobreak\iffalse
35 \let\if@noskipsec\iffalse
36 \let\par\@par
37 \let-\@dischyp
38 \let'\@acci\let'\@accii\let\=\@acciii

```

```

39   \parindent\z@ \parskip\z@skip
40   \everypar{}%
41   \linewidth\hsize
42   \@totalleftmargin\z@
43   \leftskip\z@skip \rightskip\z@skip \@rightskip\z@skip
44   \parfillskip\@flushglue
45   \lineskip\normallineskip
46   \lineskiplimit\normallineskiplimit
47   \baselineskip\normalbaselineskip
48   \sloppy}%

```

Then we redefine it by removing the setting of `\leftskip`, `\rightskip`, `\@rightskip` and `\parfillskip` and instead calling `\RaggedRight`.

```

49   \renewcommand{\@arrayparboxrestore}{%
50     \let\if@nobreak\iffalse
51     \let\if@noskipsec\iffalse
52     \let\par\@par
53     \let\-\@dischyph
54     \let'\@acci\let\'\@accii\let\=\@acciii
55     \parskip\z@skip
56     \everypar{}%
57     \linewidth\hsize
58     \@totalleftmargin\z@
59     \RaggedRight
60     \lineskip\normallineskip
61     \baselineskip\normalbaselineskip
62     \lineskiplimit\normallineskiplimit
63     \sloppy}%

```

Now we self-destroy so the command can be called more than once without causing harm (and it also frees up some space).

```

64   \let\@raggedtwoe@raggedrightboxes@opt\relax
65   }

```

Finally the declaration of the option.

```

66 \DeclareOption{raggedrightboxes}{\@raggedtwoe@raggedrightboxes@opt}

```

6.2.4 footnotes option

The option `footnotes` just sets a flag (`\if@raggedtwoe@footmisc`) to load the `footmisc` package and passes the option `ragged` to it.

```

67 \DeclareOption{footnotes}{%
68   \@raggedtwoe@footmisctrue
69   \PassOptionsToPackage{ragged}{footmisc}%
70 }

```

6.2.5 document option

The option `document` sets the complete document ragged-right by executing `\RaggedRight` via `\AtBeginDocument` and also executing the `raggedrightboxes` option.

`\@raggedtwo@abdhook` `\@raggedtwo@abdhook` is the code executed via `\AtBeginDocument`: Give a message on the terminal, execute `\RaggedRight` and self-destroy. We also make `\@tocrmarg` flexible; otherwise long lines in the table of contents (and similar tables) would not be broken because the spaceskip is rigid.

```

71 \newcommand{\@raggedtwo@abdhook}{%
72   \PackageInfo{ragged2e}{ABD: executing \string\RaggedRight}%
73   \RaggedRight
74   \@ifundefined{tocrmarg}{\edef\tocrmarg{\tocrmarg plus 2em}}%
75   \let\@raggedtwo@abdhook\relax
76 }

```

`\@raggedtwo@document@opt` `\@raggedtwo@document@opt` is the code executed via `\DeclareOption`: Insert the code into `\AtBeginDocument`, execute the `raggedrightboxes` and `footnotes` options and self-destroy.

```

77 \newcommand{\@raggedtwo@document@opt}{%
78   \AtBeginDocument{\@raggedtwo@abdhook}%
79   \@raggedtwo@raggedrightboxes@opt
80   \@raggedtwo@footmisctrue
81   \let\@raggedtwo@document@opt\relax
82 }

```

Finally the declaration of the option.

```

83 \DeclareOption{document}{\@raggedtwo@document@opt}

```

6.2.6 Other options

All unused options are passed to the `footmisc` package if the `footnotes` option is selected; otherwise the usual error is raised.

```

84 \DeclareOption*{%
85   \if@raggedtwo@footmisc
86     \PassOptionsToPackage{\CurrentOption}{footmisc}%
87   \else
88     \OptionNotUsed
89   \fi
90 }

```

6.3 Executing options

The default options are `originalcommands` and `newparameters`.

```

91 \ExecuteOptions{originalcommands,newparameters}
92 \ProcessOptions\relax

```

6.4 Loading packages

We need the `everysel` package for older kernels.

```

93 \providecommand\IfFormatAtLeastTF{\@ifl@t@r\fmtversion}
94 \IfFormatAtLeastTF{2021/01/05}{\RequirePackage{everysel}}

```

If the option `footnotes` is selected, we load the `footmisc` package after we are finished (`footmisc` detects our presence by looking for the definition of `\RaggedRight`, so we can not load it just now).

```
95 \if@raggedtwo@footmisc
96   \RequirePackage{footmisc}[2002/08/28]
97 \fi
```

6.5 Allocations

```
\CenteringLeftskip First we allocate the parameters
\RaggedLeftLeftskip 98 \newlength{\CenteringLeftskip}
\RaggedRightLeftskip 99 \newlength{\RaggedLeftLeftskip}
\CenteringRightskip 100 \newlength{\RaggedRightLeftskip}
\RaggedLeftRightskip 101 \newlength{\CenteringRightskip}
\RaggedRightRightskip 102 \newlength{\RaggedLeftRightskip}
\CenteringParfillskip 103 \newlength{\RaggedRightRightskip}
\RaggedLeftParfillskip 104 \newlength{\CenteringParfillskip}
\RaggedRightParfillskip 105 \newlength{\RaggedLeftParfillskip}
\JustifyingParfillskip 106 \newlength{\RaggedRightParfillskip}
\CenteringParindent 107 \newlength{\JustifyingParfillskip}
\RaggedLeftParindent 108 \newlength{\CenteringParindent}
\RaggedRightParindent 109 \newlength{\RaggedLeftParindent}
\JustifyingParindent 110 \newlength{\RaggedRightParindent}
111 \newlength{\JustifyingParindent}
```

6.6 Initializations

Depending on `\if@raggedtwo@originalparameters` we initialize the parameters with the values \LaTeX uses for its own commands or with our new parameters.

```
112 \if@raggedtwo@originalparameters
113   \CenteringLeftskip\@flushglue
114   \RaggedLeftLeftskip\@flushglue
115   \RaggedRightLeftskip\z@skip
116   \CenteringRightskip\@flushglue
117   \RaggedLeftRightskip\z@skip
118   \RaggedRightRightskip\@flushglue
119   \CenteringParfillskip\z@skip
120   \RaggedLeftParfillskip\z@skip
121   \RaggedRightParfillskip\@flushglue
122   \CenteringParindent\z@
123   \RaggedLeftParindent\z@
124   \RaggedRightParindent\z@
125 \else
126   \CenteringLeftskip\z@\@plus\tw@ em
127   \RaggedLeftLeftskip\z@\@plus\tw@ em
128   \RaggedRightLeftskip\z@skip
129   \CenteringRightskip\z@\@plus\tw@ em
130   \RaggedLeftRightskip\z@skip
131   \RaggedRightRightskip\z@\@plus\tw@ em
```

```

132 \CenteringParfillskip\z@skip
133 \RaggedLeftParfillskip\z@skip
134 \RaggedRightParfillskip\@flushglue
135 \CenteringParindent\z@
136 \RaggedLeftParindent\z@
137 \RaggedRightParindent\z@
138 \fi
139 \JustifyingParfillskip\@flushglue
140 \JustifyingParindent\parindent

```

6.7 Distinguishing between monospaced and proportional fonts

To set ragged text with proportional fonts *and* monospaced fonts correctly, we must distinguish between these two kinds of fonts *everytime* a font is loaded. Otherwise the settings for e. g. a proportional fonts would be in effect if you start `\RaggedRight` in `\rmfamily` and then switch to `\ttfamily`.

The goal is to have a rigid interword space in all fonts. \TeX 's interword space is `\fontdimen2` plus `\fontdimen3` minus `\fontdimen4`. This can be overwritten by setting `\spaceskip` (space between words, if nonzero) and `\xspaceskip` (space at the end of sentences, if nonzero).

We do the setting with the help of `everysel` [8], which allows us to define code which is (hopefully) executed after every fontchange in a \LaTeX document.[‡]

```

\if@raggedtwo@spaceskip \if@raggedtwo@spaceskip signals the use of commands defined by
ragged2e to the command inserted into \selectfont. It is set to true by
these commands and restored to false by  $\text{\TeX}$  when the scope of them
ends.

```

```

141 \newif\if@raggedtwo@spaceskip

```

```

\@raggedtwo@everyselectfont \@raggedtwo@everyselectfont is our code inserted into \selectfont.

```

```

142 \newcommand{\@raggedtwo@everyselectfont}{%
143 \if@raggedtwo@spaceskip

```

If no command defined by `ragged2e` is in use, we do nothing. But if it is, we look at `\fontdimen3` to see if the current font is monospaced or not.

```

144 \ifdim\fontdimen\thr@@\font=\z@\relax

```

If it is, we set `\spaceskip` to `0pt` so the interword space will be the one specified by the font designer – which is rigid anyway for monospaced fonts.

```

145 \spaceskip\z@
146 \else

```

[‡]It *is* executed after every `\selectfont`, so if you stay within NFSS and don't declare your fonts with commands like `\newfont` and then switch to them, it will work.

For proportional fonts we make the interword space rigid by setting `\spaceskip` to `\fontdimen2`.

```
147     \spaceskip\fontdimen\tw@\font
148     \fi
```

We have to reset the interword space if we are not active.

```
149     \else
150     \spaceskip\z@
151     \fi
152 }
```

If our kernel is new enough we use the kernel hook directly instead of the `everyselectfont` macro.

```
153 \IfFormatAtLeastTF{2021/01/05}
154 {\AddToHook{selectfont}{\@raggedtwoe@everyselectfont}}
155 {\EverySelectfont{\@raggedtwoe@everyselectfont}}
```

6.8 The commands

`\@raggedtwoe@savedcr` We save the definition of `\` in `\@raggedtwoe@savedcr`.

```
156 \let\@raggedtwoe@savedcr\
```

`\@raggedtwoe@saved@gnewline` We save the definition of `\@gnewline` in `\@raggedtwoe@saved@gnewline`.

```
157 \let\@raggedtwoe@saved@gnewline\@gnewline
```

`\@raggedtwoe@gnewline` The following definition of a `\@gnewline` used by the ragged commands was suggested by Markus Kohm:

```
158 \newcommand*{\@raggedtwoe@gnewline}[1]{%
159     \ifvmode
160     \@nolnerr
161     \else
162     \unskip
163     \ifmmode
```

In formulas we use the original code of `\@gnewline`.

```
164     \reserved@e {\reserved@f #1}\nobreak \hfil \break
165     \else
166     \reserved@e {\reserved@f #1}{\parskip\z@\par}%
167     \fi
168     \fi
169 }
```

`\Centering` `\Centering` first lets `\` = `\@centercr`, but only if `\` has its original meaning, otherwise `\Center` would not work inside environments like `tabular` etc., in which `\` has a different meaning. It also sets `\@gnewline` to `\@raggedtwoe@gnewline`. Then, the `LATEX` and `TEX`-parameters are set. `\@rightskip` is `LATEX`'s version of `\rightskip`.

“Every environment, like the list environments, that set `\rightskip` to its ‘normal’ value set it to `\@rightskip`” [1]

Finally we signal the code inserted into `\selectfont` that we are active and call that code directly.

```

170 \newcommand{\Centering}{%
171   \ifx\\@raggedtwo@savedcr
172     \let\\@centercr
173   \fi
174   \let\@gnewline\@raggedtwo@gnewline
175   \leftskip\CenteringLeftskip
176   \@rightskip\CenteringRightskip
177   \rightskip\@rightskip
178   \parfillskip\CenteringParfillskip
179   \parindent\CenteringParindent
180   \@raggedtwo@spaceskiptrue
181   \@raggedtwo@everyselectfont
182 }

```

`\RaggedLeft` `\RaggedLeft` is like `\Centering`; it only uses other parameters.

```

183 \newcommand{\RaggedLeft}{%
184   \ifx\\@raggedtwo@savedcr
185     \let\\@centercr
186   \fi
187   \let\@gnewline\@raggedtwo@gnewline
188   \leftskip\RaggedLeftLeftskip
189   \@rightskip\RaggedLeftRightskip
190   \rightskip\@rightskip
191   \parfillskip\RaggedLeftParfillskip
192   \parindent\RaggedLeftParindent
193   \@raggedtwo@spaceskiptrue
194   \@raggedtwo@everyselectfont
195 }

```

`\RaggedRight` `\RaggedRight` is like `\Centering`; it only uses other parameters.

```

196 \newcommand{\RaggedRight}{%
197   \ifx\\@raggedtwo@savedcr
198     \let\\@centercr
199   \fi
200   \let\@gnewline\@raggedtwo@gnewline
201   \leftskip\RaggedRightLeftskip
202   \@rightskip\RaggedRightRightskip
203   \rightskip\@rightskip
204   \parfillskip\RaggedRightParfillskip
205   \parindent\RaggedRightParindent
206   \@raggedtwo@spaceskiptrue
207   \@raggedtwo@everyselectfont
208 }

```

`\justifying` `\justifying` switches back to the defaults used by L^AT_EX for typesetting justified text.

```

209 \newcommand{\justifying}{%
210   \let\\@raggedtwoe@savedcr
211   \let@gnewline@raggedtwoe@saved@gnewline
212   \leftskip\z@
213   \@rightskip\z@
214   \rightskip\@rightskip
215   \parfillskip\JustifyingParfillskip
216   \parindent\JustifyingParindent
217   \@raggedtwoe@spaceskipfalse
218   \@raggedtwoe@everyselectfont
219 }

```

6.9 The environments

Center The environments `Center`, `FlushLeft`, and `\FlushRight` are implemented like their counterparts in \LaTeX : Start a trivlist and switch on the right command.

FlushLeft

FlushRight

```

220 \newenvironment{Center}{%
221   \trivlist
222   \Centering\item\relax
223 }{%
224   \endtrivlist
225 }
226 \newenvironment{FlushLeft}{%
227   \trivlist
228   \RaggedRight\item\relax
229 }{%
230   \endtrivlist
231 }
232 \newenvironment{FlushRight}{%
233   \trivlist
234   \RaggedLeft\item\relax
235 }{%
236   \endtrivlist
237 }

```

justify `justify` is similar to the other environments: Start a trivlist and use `\justifying`.

```

238 \newenvironment{justify}{%
239   \trivlist
240   \justifying\item\relax
241 }{%
242   \endtrivlist
243 }

```

6.10 Overloading the \LaTeX -commands

If the option `newcommands` is used, we save the original \LaTeX -commands and environments for ragged text and overload them.

```

244 \if@raggedtwoe@originalcommands

```

```

245 \else
246   \let\LaTeXcentering\centering
247   \let\LaTeXraggedleft\raggedleft
248   \let\LaTeXraggedright\raggedright
249   \let\centering\Centering
250   \let\raggedleft\RaggedLeft
251   \let\raggedright\RaggedRight
252   \let\LaTeXcenter\center
253   \let\endLaTeXcenter\endcenter
254   \let\LaTeXflushleft\flushleft
255   \let\endLaTeXflushleft\endflushleft
256   \let\LaTeXflushright\flushright
257   \let\endLaTeXflushright\endflushright
258   \let\center\Center
259   \let\endcenter\endCenter
260   \let\flushleft\FlushLeft
261   \let\endflushleft\endFlushLeft
262   \let\flushright\FlushRight
263   \let\endflushright\endFlushRight
264 \fi

265 </package>

```

7 Acknowledgements

A first version of this package for L^AT_EX 2.09 was named `raggedri` [9]. Laurent Siebenmann (lcs@topo.math.u-psud.fr) with his style `ragged.sty` [10] provided the final impulse for this new implementation.

The code for `\justifying`, `justify` and the overloading of `\@arrayparboxrestore` is incorporated from the `raggedr` [3] package by James Kilfinger (mapdn@csv.warwick.ac.uk).

Without the constant nagging of Rainer Sieger (rsieger@awi-bremerhaven.de) this package might not be.

Markus Kohm (markus.kohm@gmx.de) provided the code for `\@gnewline`.

Frank Mittelbach (frank.mittelbach@latex-project.org) provided the impetus for version 2.00.

Rolf Niepraschk (Rolf.Niepraschk@gmx.de) and Hubert Gäßlein found many bugs and provided fixes for them and code for new features.

Jordan Firth (jafirth@ncsu.edu) provided the final push for version 2.2.

References

- [1] Johannes Braams, David Carlisle, Alan Jeffrey, Leslie Lamport, Frank Mittelbach, Chris Rowley, and Rainer Schöpf. `ltxmiscn.dtx`. Part of the L^AT_EX-distribution.

- [2] Robin Fairbairns. `footmisc` — a portmanteau package for customising footnotes in $\LaTeX 2_{\epsilon}$. <https://mirror.ctan.org/macros/latex/contrib/footmisc/>.
- [3] James Kilfiger. <https://ctan.org/tex-archive/obsolete/macros/latex/contrib/misc/raggedr.sty>. $\LaTeX 2_{\epsilon}$ package.
- [4] Donald E. Knuth. *The $T_{\epsilon}X$ Book*, volume A of *Computers & Typesetting*. Addison-Wesley, Reading, MA, USA, 1986.
- [5] Leslie Lamport. *\LaTeX : A Document Preparation System*. Addison-Wesley, Reading, MA, USA, first edition, 1986.
- [6] Frank Mittelbach. The `lthooks` package. <http://mirrors.ctan.org/macros/latex/base/lthooks-doc.pdf>
- [7] Frank Mittelbach and Rainer Schöpf. The file `cmfonts.fdd` for use with $\LaTeX 2_{\epsilon}$. Part of the \LaTeX -distribution.
- [8] Martin Schröder. The obsolete `everysel`-package. <http://mirrors.ctan.org/macros/latex/contrib/everysel/everysel.pdf>. $\LaTeX 2_{\epsilon}$ package.
- [9] Martin Schröder. The `raggedri` document option. Was in <http://mirrors.ctan.org/tex-archive/macros/latex209/contrib/raggedright>. $\LaTeX 2.09$ style, outdated.
- [10] Laurent Siebenmann. `ragged.sty`. [CTAN:tex-archive/macros/generic/ragged.sty](http://ctan.org/tex-archive/macros/generic/ragged.sty). generic macro file for `plain` and \LaTeX .

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

Symbols	
<code>\'</code>	38, 54
<code>\-</code>	37, 53
<code>\=</code>	38, 54
<code>\@@par</code>	36, 52
<code>\@acci</code>	38, 54
<code>\@accii</code>	38, 54
<code>\@acciii</code>	38, 54
<code>\@arrayparboxrestore</code>	33, 49
<code>\@centercr</code>	3, 172, 185, 198
<code>\@dischyph</code>	37, 53
<code>\@flushglue</code>	4, 8, 44, 113, 114, 116, 118, 121, 134, 139
<code>\@gnewline</code> ...	157, 174, 187, 200, 211
<code>\@ifl@t@r</code>	93
<code>\@ifundefined</code>	74
<code>\@nolnerr</code>	160
<code>\@plus</code>	126, 127, 129, 131
<code>\@raggedtwoe@abdhook</code>	<u>71</u> , 78
<code>\@raggedtwoe@document@opt</code> ..	<u>77</u> , 83
<code>\@raggedtwoe@everyselectfont</code> <u>142</u> , 181, 194, 207, 218
<code>\@raggedtwoe@footmisc>true</code> ..	68, 80
<code>\@raggedtwoe@gnewline</code> <u>158</u> , 174, 187, 200
<code>\@raggedtwoe@originalcommandsfalse</code> 26, 27
<code>\@raggedtwoe@originalcommandstrue</code> 24, 25
<code>\@raggedtwoe@originalparametersfalse</code>	

- 30, 31
 - \@raggedtwo@originalparameterstrue 28, 29
 - \@raggedtwo@raggedrightboxes@opt 32, 66, 79
 - \@raggedtwo@saved@gnewline 157, 211
 - \@raggedtwo@saved@cr 156, 171, 184, 197, 210
 - \@raggedtwo@spaceskipfalse .. 217
 - \@raggedtwo@spaceskiptrue 180, 193, 206
 - \@rightskip 4, 5, 43, 176, 177, 189, 190, 202, 203, 213, 214
 - \@tocrmarg 74
 - \@totalleftmargin 42, 58
 - \\ 3, 156, 171, 172, 184, 185, 197, 198, 210
 - \‘ 38, 54
- A**
- \AddToHook 154
 - \AtBeginDocument 78
- B**
- \baselineskip 47, 61
 - \break 164
- C**
- \Center 258
 - Center (environment) 5, 220
 - \center 252, 258
 - \Centering 3, 170, 222, 249
 - \centering 246, 249
 - \CenteringLeftskip 4, 98, 113, 126, 175
 - \CenteringParfillskip 4, 98, 119, 132, 178
 - \CenteringParindent 4, 98, 122, 135, 179
 - \CenteringRightskip 4, 98, 116, 129, 176
 - \CheckCommand 33
 - \CurrentOption 86
- D**
- \DeclareOption 24, 25, 26, 27, 28, 29, 30, 31, 66, 67, 83, 84
 - \def 2, 12, 16
- E**
- \edef 74
 - \else 87, 125, 146, 149, 161, 165, 245
 - \endCenter 259
- \endcenter 253, 259
 - \endFlushLeft 261
 - \endflushleft 255, 261
 - \endFlushRight 263
 - \endflushright 257, 263
 - \endLaTeXcenter 253
 - \endLaTeXflushleft 255
 - \endLaTeXflushright 257
 - \endtrivlist 224, 230, 236, 242
- environments:
- Center 5, 220
 - FlushLeft 5, 220
 - FlushRight 5, 220
 - justify 5, 238
- \everypar 40, 56
 - \EverySelectfont 155
 - \ExecuteOptions 91
- F**
- \fi 89, 97, 138, 148, 151, 167, 168, 173, 186, 199, 264
 - \FlushLeft 260
 - FlushLeft (environment) 5, 220
 - \flushleft 254, 260
 - \FlushRight 262
 - FlushRight (environment) 5, 220
 - \flushright 256, 262
 - \fmtversion 93
 - \font 144, 147
 - \fontdimen 144, 147
- H**
- \hfil 164
 - \hsize 41, 57
- I**
- \if@nobreak 34, 50
 - \if@noskipsec 35, 51
 - \if@raggedtwo@footmisc 23, 85, 95
 - \if@raggedtwo@originalcommands 21, 244
 - \if@raggedtwo@originalparameters 22, 112
 - \if@raggedtwo@spaceskip . 141, 143
 - \ifdim 144
 - \iffalse 34, 35, 50, 51
 - \IfFormatAtLeastTF 93, 94, 153
 - \ifmmode 163
 - \ifvmode 159
 - \ifx 171, 184, 197
 - \item 222, 228, 234, 240
- J**
- justify (environment) 5, 238
 - \justifying 3, 209, 240

- `\JustifyingParfillskip`
 4, 98, 139, 215
`\JustifyingParindent` 4, 98, 140, 216
- L**
- `\LaTeXcenter` 252
`\LaTeXcentering` 246
`\LaTeXflushleft` 254
`\LaTeXflushright` 256
`\LaTeXraggedleft` 247
`\LaTeXraggedright` 248
`\leftskip` . 6, 43, 175, 188, 201, 212
`\let` 3, 34,
 35, 36, 37, 38, 50, 51, 52, 53,
 54, 64, 75, 81, 156, 157, 172,
 174, 185, 187, 198, 200, 210,
 211, 246, 247, 248, 249, 250,
 251, 252, 253, 254, 255, 256,
 257, 258, 259, 260, 261, 262, 263
`\lineskip` 45, 60
`\lineskiplimit` 46, 62
`\linewidth` 41, 57
- N**
- `\newcommand` 32, 71,
 77, 142, 158, 170, 183, 196, 209
`\newenvironment` . 220, 226, 232, 238
`\newif` 21, 22, 23, 141
`\newlength` 98,
 99, 100, 101, 102, 103, 104,
 105, 106, 107, 108, 109, 110, 111
`\nobreak` 164
`\normalbaselineskip` 47, 61
`\normallineskip` 45, 60
`\normallineskiplimit` 46, 62
- O**
- `\OptionNotUsed` 88
- P**
- `\PackageInfo` 72
`\par` 36, 52, 166
`\parfillskip` . 44, 178, 191, 204, 215
`\parindent`
 .. 7, 39, 140, 179, 192, 205, 216
`\parskip` 39, 55, 166
`\PassOptionsToPackage` 69, 86
`\ProcessOptions` 92
`\providecommand` 93
- R**
- `\RaggedLeft` 3, 183, 234, 250
`\raggedleft` 247, 250
- `\RaggedLeftLeftskip`
 4, 98, 114, 127, 188
`\RaggedLeftParfillskip`
 4, 98, 120, 133, 191
`\RaggedLeftParindent`
 4, 98, 123, 136, 192
`\RaggedLeftRightskip`
 4, 98, 117, 130, 189
`\RaggedRight`
 ... 3, 59, 72, 73, 196, 228, 251
`\raggedright` 2, 12, 248, 251
`\RaggedRightLeftskip`
 4, 98, 115, 128, 201
`\RaggedRightParfillskip`
 4, 98, 121, 134, 204
`\RaggedRightParindent`
 4, 98, 124, 137, 205
`\RaggedRightRightskip`
 4, 98, 118, 131, 202
`\relax` 15, 18, 64, 75,
 81, 92, 144, 222, 228, 234, 240
`\renewcommand` 49
`\RequirePackage` 94, 96
`\reserved@e` 164, 166
`\reserved@f` 164, 166
`\rightskip`
 5, 13, 18, 43, 177, 190, 203, 214
- S**
- `\sloppy` 48, 63
`\spaceskip` 14, 145, 147, 150
`\string` 72
- T**
- `\thr@@` 144
`\trivlist` 221, 227, 233, 239
`\tt` 17
`\ttraggedright` 16
`\tw@` 126, 127, 129, 131, 147
- U**
- `\unskip` 162
- X**
- `\xspaceskip` 15
- Z**
- `\z@` 6, 7, 13,
 18, 39, 42, 58, 122, 123, 124,
 126, 127, 129, 131, 135, 136,
 137, 144, 145, 150, 166, 212, 213
`\z@skip` 39, 43, 55, 115,
 117, 119, 120, 128, 130, 132, 133

Change History

v1.00	
General: New from raggedright V 1.21	1
v1.01	
General: Documentation improved	1
v1.02	
General: Moved to LPPL	1
v2.00	
<code>\@raggedtwoe@abdhook</code> : New macro	9
<code>\@raggedtwoe@document@opt</code> : New macro	9
<code>\@raggedtwoe@everyselectfont</code> : Completely redesigned and removed	
<code>\RaggedSpaceskip</code> and <code>\RaggedXSpaceskip</code>	11
<code>\@raggedtwoe@gnewline</code> : New macro	12
<code>\@raggedtwoe@raggedrightboxes@opt</code> : New macro	7
<code>\@raggedtwoe@saved@gnewline</code> : New macro	12
<code>\Centering</code> : Call <code>\@raggedtwoe@everyselectfont</code> and switch	
<code>\@gnewline</code>	13
<code>\JustifyingParfillskip</code> : New macro	10
<code>\JustifyingParindent</code> : New macro	10
<code>\RaggedLeft</code> : Call <code>\@raggedtwoe@everyselectfont</code> and switch	
<code>\@gnewline</code>	13
<code>\RaggedRight</code> : Call <code>\@raggedtwoe@everyselectfont</code> and switch	
<code>\@gnewline</code>	13
<code>\if@raggedtwoe@footmisc</code> : New macro	7
<code>\justifying</code> : New macro	13
General: Allow all-lowercase versions of options	7
Allow all-lowercase versions of options and removed documentation of mixed-case versions.	5
Incorporated <code>raggedr</code>	1
Load the <code>footmisc</code> package	9
New command <code>\justifying</code>	3
New environment <code>justify</code>	5
New option <code>document</code>	6, 8
New option <code>footnotes</code>	6, 8
New option <code>raggedrightboxes</code>	6, 7
Pass all other options to <code>footmisc</code> if it's loaded	9
Removed <code>\RaggedSpaceskip</code> and <code>\RaggedXSpaceskip</code>	3, 10
Removed spaces and unneeded braces from <code>\setlength</code> ; replaced <code>plus</code> with <code>\@plus</code>	6
<code>justify</code> : New environment	14
v2.01	
<code>\@raggedtwoe@everyselectfont</code> : Removed the setting of <code>\xspaceskip</code>	11
v2.02	
<code>\@raggedtwoe@gnewline</code> : Bugfix: <code>\@nolerr</code> → <code>\@nolnerr</code>	12
General: Bugfix: <code>\if@raggedtwoe@footmisc>true</code> →	
<code>\@raggedtwoe@footmisc>true</code>	8
Removed <code>\setlength</code>	6
Use <code>\@flushglue</code>	6
<code>FlushRight</code> : Use <code>\trivlist</code> ... <code>\endtrivlist</code> instead of	
<code>\begin{trivlist}</code> ... <code>\end{trivlist}</code>	14
<code>justify</code> : Use <code>\trivlist</code> ... <code>\endtrivlist</code> instead of <code>\begin{trivlist}</code> ... <code>\end{trivlist}</code>	14

v2.03		
	General: Bugfix: footnotes was actually raggedrightboxes	8
v2.04		
	<code>\@raggedtwoe@abdhook</code> : Set <code>\@tocrmarg</code> and use <code>\PackageInfo</code>	9
	<code>\@raggedtwoe@everyselectfont</code> : Reset <code>\spaceskip</code> when we are not active	12
	<code>\@raggedtwoe@raggedrightboxes@opt</code> : The setting of <code>\parindent</code> is superfluous	8
	General: Initialize <code>\JustifyingParindent</code> with <code>\parindent</code>	10
	Insert missing <code>\</code>	10
	Save more commands	14
v2.1		
	<code>\@raggedtwoe@abdhook</code> : bugfix: Use <code>\@tocrmarg</code> only if it's defined	9
	<code>\@raggedtwoe@gnewline</code> : Bugfix: handle math	12
	General: bugfix: <code>\Flushleft</code> instead of <code>\FlushLeft</code> (found by Berend Hasselman)	14
	bugfix: Load <code>footmisc</code> directly and not via <code>\AtEndOfPackage</code> (bug found by Axel Sommerfeldt)	10
	document that document needs <code>footmisc</code>	6
v2.2		
	<code>\@raggedtwoe@raggedrightboxes@opt</code> : Definition of <code>\@arrayparboxrestore</code> has changed	7
	General: Change maintenance status of package	2
	Move to git/gitlab, use <code>svninfo2</code>	1
	Require a new version of L ^A T _E X(2017/03/29)	1
v3.0		
	General: Change maintenance status	2
	document <code>everyssel</code> is obsolete	6
v3.00		
	<code>\@raggedtwoe@everyselectfont</code> : Use kernel hook if available	12
	General: Remove the <code>everyssel</code> package if kernel hooks are available	9