



# The bidi Package

User documentation for version 39.1

to my Master, *Ferdowsi The Great*

If you want to report any bugs or typos and corrections in the documentation, or ask for any new features, or suggest any improvements, or ask any questions about the package, then please use the issue tracker:

<https://github.com/kvafa/bidi/issues>

In doing so, please always explain your issue well enough, and always include a minimal working example showing the issue.

You may also have conversations, ask questions and post answers without opening issues using the Discussions space:

<https://github.com/kvafa/bidi/discussions>

The announcements for the new releases of the package will also appear in the Discussions space under the Announcements category.

Copyright (c) 2007–2022 Vafa Khalighi  
Copyright (c) 2018–2020 bidi-tex GitHub Organization

Permission is granted to distribute and/or modify *both the documentation and the code* under the conditions of the L<sup>A</sup>T<sub>E</sub>X Project Public License, either version 1.3c of this license or (at your option) any later version.

# The *bid*i Package

## Bidirectional typesetting in plain $\text{T}_{\text{E}}\text{X}$ and $\text{\LaTeX}$ , using $\text{XeT}_{\text{E}}\text{X}$ engine

User documentation for version 39.1

Vafa Khalighi\*

July 12, 2022

## Contents

<b>I</b>	<b><math>\text{\LaTeX}</math> Manual</b>	<b>3</b>		
<b>1</b>	<b>Basics</b>	<b>3</b>		
1.1	Loading The Package . . . . .	3	1.19	LTRbibitems and RTLbibitems Environments . . . . . 12
1.2	<i>bid</i> i's Symbol . . . . .	4	1.20	<code>\setLTRbibitems</code> , <code>\setRTLbibitems</code> , and <code>\setdefaultbibitems</code> commands . . . . . 12
1.3	Commands for Version number, and Date of The Package . . . . .	4	1.21	Typesetting margin par . . . . . 12
1.4	Turning $\text{TeX}$ -- $\text{XeT}$ features on and off . . . . .	4	1.22	Typesetting of Headers and Footers . . . . . 13
1.5	Options of The Package . . . . .	4	1.23	Tabular Typesetting . . . . . 13
1.6	Paragraph Switching Commands	5	1.24	Equation Numbers . . . . . 13
1.7	Paragraph Switching Environments	5	<b>2</b>	<b>Support For Various Packages and Classes</b> <b>13</b>
1.8	Typesetting Short LTR and RTL Texts . . . . .	6	2.1	Color . . . . . 14
1.9	Footnotes . . . . .	7	2.2	The <code>dblfnote</code> package . . . . . 14
1.9.1	Footnote Rule . . . . .	7	2.3	Hyperref . . . . . 14
1.10	Two Column Typesetting . . . . .	8	2.4	The <code>listings</code> package . . . . . 14
1.11	RTL cases . . . . .	8	2.5	<code>flowfram</code> Package . . . . . 15
1.12	Typesetting Logos . . . . .	8	2.6	Multicolumn Typesetting . . . . . 15
1.13	Separation Mark . . . . .	9	<b>3</b>	<b>Extra <i>bid</i>i Packages and Classes</b> <b>15</b>
1.14	<code>\raggedright</code> , <code>\raggedleft</code> commands, <code>flushleft</code> and <code>flushright</code> Environments . . . . .	9	3.1	<code>biditufte</code> bundle . . . . . 15
1.15	Primitive-like commands . . . . .	9	3.2	Typesetting $\text{T}_{\text{E}}\text{X}$ and $\text{\LaTeX}$ Codes
1.16	Something To know about <code>\hbox</code>	10	3.3	Typesetting Poems . . . . . 16
1.17	<code>\bidillap</code> and <code>\bidirlap</code> Commands . . . . .	10	3.4	Typesetting Résumé . . . . . 16
1.18	LTRitems and RTLitems Envi- ronments . . . . .	10	3.5	Print Two Pages On A Single Page
			3.6	Producing Presentations . . . . . 17

---

\*Thanks to the *bid*i-tex GitHub organization consisted of David Carlisle and other members of the  $\text{\LaTeX}$  team to maintain the package in the period 2018–2020.

<b>4</b>	<b>Some Useful Internal Macros and Programming Tips</b>	<b>17</b>			
4.1	Equating Conditionals . . . . .	17	5.2	Commands for Version number, and Date of The Package . . . . .	19
4.2	RTL Conditional . . . . .	17	5.3	Turning TeX--XeT features on and off . . . . .	19
4.3	Main RTL Conditional . . . . .	17	5.4	Paragraph Switching Commands	20
4.4	Latin Conditional . . . . .	17	5.5	Paragraph Switching Environments	20
4.5	Tags Internal Macro . . . . .	17	5.6	Typesetting Short LTR and RTL Texts . . . . .	20
4.6	Definition File Loaded Internal Macro . . . . .	18	5.7	Primitive-like commands . . . . .	21
4.7	Tabular Conditional . . . . .	18	5.8	Something To know about \hbox	21
4.8	Footnote Conditional . . . . .	18	5.9	Typesetting Logos . . . . .	22
4.9	Direction Ensuring Macros . . . . .	18			
4.10	Reset Direction Macro . . . . .	18	<b>6</b>	<b>Some Useful Internal Macros and Programming Tips</b>	<b>22</b>
			6.1	RTL Conditional . . . . .	22
<b>II</b>	<b>Plain TeX Manual</b>	<b>19</b>	6.2	Main RTL Conditional . . . . .	22
<b>5</b>	<b>Basics</b>	<b>19</b>	6.3	Direction Ensuring Macros . . . . .	22
5.1	Loading The Package . . . . .	19	6.4	Reset Direction Macro . . . . .	22

## Introduction

bidi provides a convenient interface for typesetting bidirectional texts in plain TeX and XeTeX, using the XeTeX engine.

## bidi Info On The Terminal and In The Log File

If you use bidi package to write any input TeX document, and then run xelatex on your document, bidi writes some information about itself to the terminal and to the log file. The information is something like:

```
*****
*
* bidi package (Bidirectional typesetting in
* plain TeX and LaTeX, using XeTeX engine)
*
* Description: A convenient interface for
* typesetting bidirectional texts in plain TeX
* and LaTeX. The package includes adaptations
* for use with many other commonly-used packages.
*
* Copyright (c) 2007--2022 Vafa Khalighi
* Copyright (c) 2018--2020 bidi-tex GitHub Organization
*
* v39.1, 2022/07/12
*
* License: LaTeX Project Public License, version
* 1.3c or higher (your choice)
*
* Location on CTAN: /macros/xetex/latex/bidi
*
* Issue tracker: https://github.com/kvafa/bidi/issues
*
```



```
* Discussions: https://github.com/kvafa/bidi/discussions
*
* Primary author of bidi package: Vafa Khalighi.
*
*****
```

## Part I

# L<sup>A</sup>T<sub>E</sub>X Manual

## 1 Basics

### 1.1 Loading The Package

You can load the package in the ordinary way;

```
\usepackage [Options] {bidi}
```

Where options of the package are explained later in [subsection 1.5](#).

When loading the package, it is important to know that:

- ❶ bidi should be the last package that you load, because otherwise you are certainly going to overwrite bidi's definitions and consequently, you will not get the expected output.
- ❷ In fact, bidi makes sure that some specific packages are loaded before bidi; these are those packages that bidi modifies them for bidirectional typesetting.

If you load bidi before any of these packages, then you will get an error saying that you should load bidi as your last package.

For instance, consider the following minimal example:

```
1 \documentclass{minimal}
2 \usepackage{bidi}
3 \usepackage{graphicx}
4 \begin{document}
5 This is just a test.
6 \end{document}
```

Where graphicx is loaded after bidi. If you run xelatex on this document, you will get an error which looks like this:

```
! Package bidi Error: Oops! you have loaded package graphicx after
bidi package. Please load package graphicx before bidi package, and
then try to run xelatex on your document again.
```

See the bidi package documentation for explanation.

Type H <return> for immediate help.

...

```
1.4 \begin{document}
```

?



## 1.2 bidi's Symbol

As you may know lion symbolizes  $\text{\TeX}$  but lion does not symbolizes bidi. **Simorgh**<sup>1</sup> (shown on the first page of this documentation) symbolizes bidi.

## 1.3 Commands for Version number, and Date of The Package

`\bidiversion \bididate`

☞ `\bidiversion` gives the current version of the package.

☞ `\bididate` gives the current date of the package.

```
1 \documentclass{article}
2 \usepackage{bidi}
3 \begin{document}
4 This is typeset by \textsf{bidi} package, \bidiversion, \bididate.
5 \end{document}
```

## 1.4 Turning TeX--XeT features on and off

The bidirectional typesetting in Xe $\text{\TeX}$  is based on TeX--XeT algorithm and bidi package automatically turns the feature on for typesetting bidirectional texts. As the internal data structures built by TeX--XeT differ from those built by  $\text{\TeX}$ , the typesetting of a document by TeX--XeT may therefore differ from that performed by  $\text{\TeX}$ . So you may notice that some packages behave differently when TeX--XeT feature is on and you may want to turn the feature off for a part of the text so that you get the default behaviour of original  $\text{\TeX}$ . Two commands are provided for this purpose:

`\TeXXeTOn \TeXXeTOff`

☞ `\TeXXeTOn` turns TeX--XeT feature on, which is active by default when bidi package is loaded.

☞ `\TeXXeTOff` turns TeX--XeT feature off.

## 1.5 Options of The Package

There are three options, namely `RTLdocument` and `rldocument`, which are essentially equivalent. If you pass any of these options to the package, you will be typesetting a document containing mainly RTL texts with some LTR texts. These options activate `\setRTL` (explained in subsection 5.4), `\RTLdblcol` (explained in subsection 1.10) and `\autofootnoterule` (explained in subsubsection 1.9.1).

It is clear that if you do not pass any of these options to the package, you will be typesetting a document containing mainly LTR texts with some RTL texts.

There is also `extrafootnotefeatures` option that allows footnotes to be typeset in different formats:

---

<sup>1</sup>**Simorgh** is an Iranian benevolent, mythical flying creature which has been shown on the titlepage of this documentation. For more details see <http://en.wikipedia.org/wiki/Simurgh>



```

\normalfootnotes
\twocolumnfootnotes \threecolumnfootnotes \fourcolumnfootnotes
\fivecolumnfootnotes \sixcolumnfootnotes \sevencolumnfootnotes
\eightcolumnfootnotes \ninecolumnfootnotes \tencolumnfootnotes
\paragraphfootnotes

```

☞ `\normalfootnotes` typesets footnotes in Standard  $\text{\LaTeX}$  format.

☞ `\twocolumnfootnotes` to `\tencolumnfootnotes`, typeset footnotes in two-columns to ten-columns, respectively.

☞ `\paragraphfootnotes` typesets footnotes in paragraph format.

Please note that when using `extrafootnotefeatures` option, the footnote rule will be as wide as the text width and `\autofooterule`, `\rightfootnoterule`, `\leftfootnoterule`, and `\textwidthfootnoterule` commands have no effects.

Please also note that if you redefine `\baselinestretch` command or change the value of `\baselineskip` primitive before `\paragraphfootnotes` command, then you may get Arithmetic Overflow error. You should change these after using `\paragraphfootnotes` command.

## 1.6 Paragraph Switching Commands

```

\setLTR \setLR \unsetRL \unsetRTL
\setRTL \setRL \unsetLTR

```

☞ With any of the commands in the first row, you can typeset LTR paragraphs.

☞ With any of the commands in the second row, you can typeset RTL paragraphs.

```

1 \documentclass{article}
2 \usepackage{bidi}
3 \begin{document}
4 \setRTL%
5 Anyone who reads Old and Middle English literary texts will be
6 familiar with the mid-brown volumes of the EETS, with the symbol
7 of Alfred's jewel embossed on the front cover.
8
9 \setLTR% Notice the blank line before \setLTR
10 Anyone who reads Old and Middle English literary texts will be
11 familiar with the mid-brown volumes of the EETS, with the symbol
12 of Alfred's jewel embossed on the front cover.
13 \end{document}

```

## 1.7 Paragraph Switching Environments

```

\begin{LTR} <text> \end{LTR}
\begin{RTL} <text> \end{RTL}

```

☞ With LTR environment, you can typeset LTR paragraphs.



☞ With RTL environment, you can typeset RTL paragraphs.

```
1 \documentclass{article}
2 \usepackage{bidi}
3 \begin{document}
4 \begin{RTL}
5 Anyone who reads Old and Middle English literary texts will be familiar
6 with the mid-brown volumes of the EETS, with the symbol
7 of Alfred's jewel embossed on the front cover.
8 \begin{LTR}
9 Anyone who reads Old and Middle English literary texts will be familiar
10 with the mid-brown volumes of the EETS, with the symbol
11 of Alfred's jewel embossed on the front cover.
12 \end{LTR}
13 And we are still typesetting RTL.
14 \end{RTL}
15 \end{document}
```

## 1.8 Typesetting Short LTR and RTL Texts

<code>\LRE{&lt;text&gt;}</code>	<code>\LR{&lt;text&gt;}</code>
<code>\RLE{&lt;text&gt;}</code>	<code>\RL{&lt;text&gt;}</code>

☞ With any of the commands in the first row, you can typeset short LTR text inside RTL paragraphs.

☞ With any of the commands in the second row, you can typeset short RTL text inside LTR paragraphs.

```
1 \begin{document}
2 \begin{RTL}
3 Anyone who reads Old and Middle English \LRE{Short LTR text} literary
4 texts will be familiar
5 with the mid-brown volumes of the EETS, with the symbol
6 of Alfred's jewel embossed on the front cover.
7 \begin{LTR}
8 Anyone who reads Old and Middle English \RLE{Short RTL text} literary
9 texts will be familiar
10 with the mid-brown volumes of the EETS, with the symbol
11 of Alfred's jewel embossed on the front cover.
12 \end{LTR}
13 \end{RTL}
14 \end{document}
```



## 1.9 Footnotes

```
\footnote [num] {<text>} \LTRfootnote [num] {<text>} \RTLfootnote [num] {<text>}  
\setfootnoteRL \setfootnoteLR \unsetfootnoteRL  
\thanks{<text>} \LTRthanks{<text>} \RTLthanks{<text>}
```

- ☞ `\footnote` in RTL mode produces an RTL footnote while in LTR mode it produces an LTR footnote.
- ☞ `\LTRfootnote` will always produce an LTR footnote, independent on the current mode.
- ☞ `\RTLfootnote` will always produce an RTL footnote, independent on the current mode.
- ☞ Specifying a `\setfootnoteRL` command anywhere will make `\footnote` produce an RTL footnote.
- ☞ Specifying either a `\setfootnoteLR` or an `\unsetfootnoteRL` command anywhere will make `\footnote` produce an LTR footnote.
- ☞ `\thanks` (to be used only inside `\author` or `\title` argument) in RTL mode produces an RTL footnote while in LTR mode it produces an LTR footnote.
- ☞ `\LTRthanks` (to be used only inside `\author` or `\title` argument) will always produce an LTR footnote, independent on the current mode.
- ☞ `\RTLthanks` (to be used only inside `\author` or `\title` argument) will always produce an RTL footnote, independent on the current mode.

```
\footnotetext [num] {<text>} \LTRfootnotetext [num] {<text>}  
\RTLfootnotetext [num] {<text>}
```

- ☞ `\footnotetext` used in conjunction with `\footnotemark`, in RTL mode produces an RTL footnote while in LTR mode it produces an LTR footnote.
- ☞ `\LTRfootnotetext` used in conjunction with `\footnotemark`, will always produce an LTR footnote, independent on the current mode.
- ☞ `\RTLfootnotetext` used in conjunction with `\footnotemark`, will always produce an RTL footnote, independent on the current mode.

### 1.9.1 Footnote Rule

The behavior of footnote rules can also be controlled.

```
\autofooterule \rightfootnoterule \leftfootnoterule  
\LRfootnoterule \textwidthfootnoterule \SplitFootnoteRule  
\debugfootnotedirection
```

- ☞ `\autofooterule` will draw the footnote rule right or left aligned based on the direction of the first footnote following the rule (i.e., put in the current page).



- ☞ `\rightfootnoterule` will put footnote rule on the right-hand side.
- ☞ `\leftfootnoterule` or `\LRfootnoterule` will put footnote rule on the left-hand side.
- ☞ `\textwidthfootnoterule` will draw the footnote rule with a width equal to `\textwidth`.
- ☞ `\SplitFootnoteRule` puts a full-width rule above the split-off part of a split footnote.
- ☞ `\debugfootnotedirection` writes the direction of the first footnote on each page, in the log file.

## 1.10 Two Column Typesetting

`\RTLdblcol   \LTRdblcol`

If you pass the `twocolumn` option to the class file and if the main direction of the document is RTL, then you get RTL two column and if the main direction of the document is LTR, then you get LTR two column. In addition, `\RTLdblcol` allows you to have RTL two column typesetting and `\LTRdblcol` allows you to have LTR two column typesetting as the options of the class file.

Also please note that in `twocolumn` documents, the width of the `\footnoterule` will be equal to `\columnwidth` no matter which footnote-rule commands you use; indeed, in `twocolumn` documents only `\textwidthfootnoterule` is active and other footnote-rule commands will not be effective.

`\RTLcases` command was previously known as `\rcases` command but since there was a clash with `mathtools` package (`mathtools` defines `rcases` environment), we had to rename `\rcases` command to `\RTLcases` command.

### 1.11 RTL cases

`\RTLcases{\text{<branch1>}\cr\text{<branch2>}\cr\text{<branch3>}...}  
\text{<main>}`

`\RTLcases` is defined in `bidi` for typesetting RTL cases. `\text` is defined in `amsmath` package, so this means that you need to load `amsmath` package too.

```

1 \documentclass{article}
2 \usepackage{amsmath}
3 \usepackage{bidi}
4 \begin{document}
5 \setRTL
6 \[ \RTLcases{\text{men}\cr\text{women}}
7 \text{Humans Beings}
8 \]
9 \end{document}

```

### 1.12 Typesetting Logos

`\XeTeX   \XeLaTeX`

`bidi` defines `XeTeX` and `XeLaTeX` logos and in addition, it makes sure that logos, `TEX`, `LATEX`, `LATEX 2ε` are typeset LTR.



### 1.13 Separation Mark

`\SepMark{<mark>} \@SepMark`

Generally in Standard  $\text{\LaTeX}$ , dot is used for separation between section numbers, equation numbers any anything else which needs to be seperated. You can use `\SepMark` to use any other mark as the seperation mark instead a dot.

```
1 \documentclass{article}
2 \usepackage{bidi}
3 \SepMark{-}
4 \begin{document}
5 \section{First}
6 \subsection{Second}
7 \subsubsection{Third}
8 \end{document}
```

If you decide to change the numbering of chapters, sections, subsections, equations, figures and ..., you should either load `amsmath` package and use `\numberwithin` macro to do this or do the ordinary way, but instead dot write `\@SepMark`. Using dot instead `\@SepMark` will certainly make trouble.

```
1 \documentclass{article}
2 \usepackage{bidi}
3 \SepMark{-}
4 \makeatletter
5 \renewcommand\theequation{\thesection\@SepMark\@arabic\c@equation}
6 \makeatother
7 \begin{document}
8 \section{First}
9 \begin{equation}
10 x^2+y^2=z^2
11 \end{equation}
12 \end{document}
```

### 1.14 `\raggedright`, `\raggedleft` commands, `flushleft` and `flushright` Environments

`\raggedright` command and `flushleft` environment put the text on the left hand side and `\raggedleft` command and `flushright` environment put the text on the right hand side, independent on the current mode.

### 1.15 Primitive-like commands

`\hboxR \hboxL \vboxR \vboxL`

☞ The syntax of `\hboxR` is exatly the same as the syntax of `\hbox`, but its contents is always typeset RTL.

☞ The syntax of `\hboxL` is exatly the same as the syntax of `\hbox`, but its contents is always typeset LTR.



☞ The syntax of `\vboxR` is exactly the same as the syntax of `\vbox`, but its contents is always typeset RTL.

☞ The syntax of `\vboxL` is exactly the same as the syntax of `\vbox`, but its contents is always typeset LTR.

### 1.16 Something To know about `\hbox`

If you enable RTL typesetting and typeset an horizontal box at the beginning of the document:

```
1 \documentclass{article}
2 \usepackage{bidi}
3 \setRTL
4 \begin{document}
5 \hbox{This is a Test}
6 \end{document}
```

You see that even you have used `\setRTL`, the horizontal box appears LTR (It appears on the left hand side and its content is typeset left to right). This is because when `TEX` starts, it is in the vertical mode so if you need to have that `\hbox` appear RTL, then write `\leavevmode` before `\hbox`:

```
1 \documentclass{article}
2 \usepackage{bidi}
3 \setRTL
4 \begin{document}
5 \leavevmode\hbox{This is a Test}
6 \end{document}
```

### 1.17 `\bidillap` and `\bidirlap` Commands

In RTL mode, `\llap` and `\rlap` do opposite things. Since these two macros are used in a lot of classes and packages, instead modifying these two macros, we have created two new macros `\bidillap` and `\bidirlap` which give logical results.

### 1.18 `LTRitems` and `RTLitems` Environments

If you typeset an `itemize`, or an `enumerate`, or a `description` environment where all `\items` are one directional, you have no problem at all as shown below:

```
1 \documentclass{article}
2 \begin{document}
3 Anyone who reads Old and Middle English literary texts will be familiar
4 with the mid-brown volumes of the EETS, with the symbol of Alfred's
5 \begin{enumerate}
6 \item Anyone who reads Old and Middle English literary texts will be
7 familiar with the mid-brown volumes of the EETS, with the symbol of
8 Alfred's
9 \item Anyone who reads Old and Middle English literary texts will be
10 familiar with the mid-brown volumes of the EETS, with the symbol of
11 Alfred's
12 \end{enumerate}
13 \end{document}
```



```
8 \end{document}
```

However if the above example becomes bidirectional, as shown below:

```
1 \documentclass{article}
2 \usepackage{bidi}
3 \begin{document}
4 Anyone who reads Old and Middle English literary texts will be familiar
5 with the mid-brown volumes of the EETS, with the symbol of Alfred's
6 \begin{enumerate}
7 \item Anyone who reads Old and Middle English literary texts will be
8 familiar with the mid-brown volumes of the EETS, with the symbol of
9 Alfred's
10 \setRTL
11 \item Anyone who reads Old and Middle English literary texts will be
12 familiar with the mid-brown volumes of the EETS, with the symbol of
13 Alfred's
14 \end{enumerate}
15 \end{document}
```

Then some people may argue that this typographically does not look promising. For this purpose, RTLitems environment is provided which has the following syntax:

```
\begin{RTLitems}
  \item <text>
  ...
\end{RTLitems}
```

By using the RTLitems environment, the previous example will look like the following:

```
1 \documentclass{article}
2 \usepackage{bidi}
3 \begin{document}
4 Anyone who reads Old and Middle English literary texts will be familiar
5 with the mid-brown volumes of the EETS, with the symbol of Alfred's
6 \begin{enumerate}
7 \item Anyone who reads Old and Middle English literary texts will be
8 familiar with the mid-brown volumes of the EETS, with the symbol of
9 Alfred's
10 \begin{RTLitems}
11 \item Anyone who reads Old and Middle English literary texts will be
12 familiar with the mid-brown volumes of the EETS, with the symbol of
13 Alfred's
14 \end{RTLitems}
15 \end{enumerate}
16 \end{document}
```

Similarly, LTRitems environment is defined which has the following syntax:

```
\begin{LTRitems}
  \item <text>
  ...
\end{LTRitems}
```



### 1.19 LTRbibitems and RTLbibitems Environments

The syntax of LTRbibitems and RTLbibitems environments is exactly like the syntax of LTRitems and RTLitems environments but there are few differences:

- ☞ LTRitems and RTLitems environments should only be used for list-like environments (such as itemize, enumerate and description environments) but LTRbibitems and RTLbibitems environments should only be used for thebibliography environment.
- ☞ Clearly instead of `\item`, you have `\bibitem` inside LTRbibitems and RTLbibitems environments.

### 1.20 \setLTRbibitems, \setRTLbibitems, and \setdefaultbibitems commands

<code>\setLTRbibitems   \setRTLbibitems   \setdefaultbibitems</code>
--

- ☞ If your whole thebibliography environment is inside RTL mode, but all your `\bibitems` are LTR and you actually want to have `\bibname` to appear on the RHS, you can use `\setLTRbibitems` command before thebibliography environment.
- ☞ If your whole thebibliography environment is inside LTR mode, but all your `\bibitems` are RTL and you actually want to have `\bibname` to appear on the LHS, you can use `\setRTLbibitems` command before thebibliography environment.
- ☞ `\setdefaultbibitems` is the default, when your `\bibitems` are a mixture of LTR and RTL and it does not matter what mode (LTR or RTL) your thebibliography environment is in. Please note that you do not have to use `\setdefaultbibitems` command in this case at all.

Consider an example that your thebibliography environment is inside LTR mode and you have, say two `\bibitems`. The first `\bibitem` is LTR and the second `\bibitem` is RTL. One could typeset this senario as shown below:

```
1 \documentclass{article}
2 \usepackage{bidi}
3 \begin{document}
4 \begin{thebibliography}{99}
5 \bibitem This is the first bibitem which is LTR.
6 \begin{RTLbibitems}
7 \bibitem This is the second bibitem which is RTL.
8 \end{RTLbibitems}
9 \end{thebibliography}
10 \end{document}
```

### 1.21 Typesetting margin par

By default, in RTL mode, `\marginpar` appears on LHS and its content is typeset RTL and in LTR mode, `\marginpar` appears on RHS and its content is typeset LTR. In addition, the following commands are provided:



```

\setRTLmarginpar \setLTRmarginpar \setdefaultmarginpar
\LTRmarginpar[⟨left-text⟩]{⟨right-text⟩}
\RTLmarginpar[⟨left-text⟩]{⟨right-text⟩}

```

- ☞ `\setRTLmarginpar` always makes `\marginpar` to appear on LHS and the content of `\marginpar` is typeset RTL (this is independent of the current mode).
- ☞ `\setLTRmarginpar` always makes `\marginpar` to appear on RHS and the content of `\marginpar` is typeset LTR (this is independent of the current mode).
- ☞ `\setdefaultmarginpar` gives the default behaviour of `\marginpar` as described above.
- ☞ `\LTRmarginpar` typesets `⟨left-text⟩` and `⟨right-text⟩` always LTR.
- ☞ `\RTLmarginpar` typesets `⟨left-text⟩` and `⟨right-text⟩` always RTL.
- ☞ in RTL mode, places of `⟨left-text⟩` and `⟨right-text⟩` swaps.

## 1.22 Typesetting of Headers and Footers

If the main direction of the document is RTL, then headers and footers are typeset RTL and if the main direction of the document is LTR, then headers and footers are typeset LTR.

## 1.23 Tabular Typesetting

In RTL mode, tabular are typeset RTL and in LTR mode, tabular are typeset LTR.

## 1.24 Equation Numbers

For `reqno`, equation numbers are on the right hand side and for `leqno`, equation numbers are on the left hand side, independent on the current mode.

# 2 Support For Various Packages and Classes

The `bidi` package supports `amsmath`, `amstext`, `amsthm`, `array`, `arydshln`, `breqn`, `cals`, `caption`, `color`, `colortbl`, `crop`, `cuted`, `cutwin`, `dblfnote`, `draftwatermark`, `empheq`, `fancyhdr`, `fancybox`, `fix2col`, `float`, `floatrow`, `flowfram`, `framed`, `ftnright`, `geometry`, `graphicx`, `hvfloa`, `hyperref`, `lettrine`, `listings`, `mdframed`, `midfloat`, `minitoc`, `multicol`, `multienum`, `newfloat`, `pdfpages`, `pstricks`, `quotchap`, `pycinpar`, `ragged2e`, `rotating`, `sidecap`, `stabular`, `subfig`, `subfigure`, `supertabular`, `xtab`, `tbls`, `tabulary`, `PGF & TIKZ`, `tocbibind`, `tocloft`, `tocstyle`, `wrapfig`, `xcolor`, `xltxtra` packages, `amsart`, `amsbook`, `artikel1`, `artikel2`, `artikel3`, `extarticle`, `flashcards`, `standrad article`, `boek`, `boek3`, `standard book`, `bookest`, `extbook`, `extletter`, `scrlettr`, `standard letter`, `memoir`, `extreport`, `rapport1`, `rapport3`, `refrep`, `standard report`, `scrartcl`, `scrbook`, `scrreprt` classes and any other packages and classes that relies on these packages and classes. This means, you can use all these packages and classes in addition to other packages and classes that rely on these packages and classes and use their functionality fully for your bidirectional documents.

We now give some details that you should know about the supported packages or classes.



## 2.1 Color

You can use `color` and `xcolor` packages to typeset texts in colours and colour boxes produced by `\colorbox` and `\fcolorbox` commands. Please note that your Coloured text should not span more than a line, if your text spans more than a line, you will be in trouble which means your whole document, page or paragraph may be coloured. If your texts spans more than a line, then you should use `xcolor` package.

Also if you are going to use `\color` command to colour the text at the beginning of a paragraph, then you should have `\leavevmode` before `\color` command.

For having coloured tabular, you can use `colortbl` package.

## 2.2 The `dblfnote` package

The `dblfnote` package makes footnotes double-columned. In addition `bidi` package adds bidirectional support for the `dblfnote` package by providing the following commands:

<code>\RTLdfnmakecol</code>	<code>\LTRdfnmakecol</code>
-----------------------------	-----------------------------

☞ `\RTLdfnmakecol` makes footnotes double-columned RTL.

☞ `\LTRdfnmakecol` makes footnotes double-columned LTR.

☞ If the main direction of the document is RTL, `\RTLdfnmakecol` is active and if the main direction of the document is LTR, `\LTRdfnmakecol` is active.

Please note that when using `dblfnote` package, the footnote rule will be as wide as the footnote column and `\autofootnoterule`, `\rightfootnoterule`, `\leftfootnoterule`, and `\textwidthfootnoterule` commands have no effects.

## 2.3 Hyperref

The `hyperref` package works fine with bidirectional documents if and only if, your link will not span more than a line. If your link spans more than a line, then your whole document, or page or paragraph may be linked.

## 2.4 The `listings` package

Suppose that the main direction of the document is RTL, and you want to typeset a piece of C code in LTR mode but you also have the requirement that the caption of that particular code should be RTL. Since the direction of the code is LTR, then the direction of the caption also will be LTR and this is not something you want; sadly there is no easy way to do this. For this very reason, the `bidi` package adds a key to the `listings` package, namely `captiondirection`. This key can have the following values:

☞ `textdirection` follows the direction of the text (default).

☞ RTL sets the direction of the caption to RTL.

☞ LTR sets the direction of the caption to LTR.

Giving any other values gives you an error.



## 2.5 flowfram Package

You can use `flowfram` package for your bidirectional documents. Please note that `flowfram` package provides support for bidirectional column typesetting, for details, see its manual.

## 2.6 Multicolumn Typesetting

In the previous versions of `bidi` package, it was recommended that you need to use `fmultico` package instead the original `multicol` package for RTL multicolumn typesetting. This is not the case any more and you should not use buggy `fmultico` package any more. Simply load the original `multicol` package before loading `bidi`. `bidi` now supports `multicol` package and you can typeset bidirectional multi columns. When using `multicol` package, the following two additional commands are provided:

`\RTLmulticolcolumns`   `\LTRmulticolcolumns`

If the main direction of the document is RTL, then you get RTL multi columns and if the main direction of the document is LTR, then you get LTR multi columns. In addition, `\RTLmulticolcolumns` allows you to have RTL multi columns and `\LTRmulticolcolumns` allows you to have LTR multi columns.

In addition, you also can use `vwcol` package for variable width bidirectional column typesetting.

## 3 Extra bidi Packages and Classes

### 3.1 bidentufte bundle

A modified version of `tufte-latex`, `bidentufte` bundle, mainly for RTL typesetting, is provided. If you never used `bidentufte` bundle or `tufte-latex` package and you want to use `bidentufte` bundle, then you need to look at `tufte-latex` package's manual and examples. In addition, for using `bidentufte` bundle, you need to know the following notes:

☞ You need to use `bidentufte-book` class instead `tufte-book` class and `bidentufte-handout` class instead `tufte-handout` class.

☞ `bidentufte` bundle provides the following extra commands:

`\LTRsidenote`   `\RTLsidenote`   `\LTRmarginnote`   `\RTLmarginnote`

☞ `bidentufte` bundle unlike `tufte-latex` package, only provides justified lines.

☞ Some features of `tufte-latex` that does not make any sense in RTL, do not exist in `bidentufte` bundle (no need for `soul`, `letterspace` and `macrotype` packages).

☞ If you want to configure `bidentufte-book` class for your own needs, then you can create a file with the name `bidentufte-book.cfg` and put your  $\TeX$  macros in that file; similarly, if you want to configure `bidentufte-handout` class for your own needs, then you can create a file with the name `bidentufte-handout.cfg` and put your  $\TeX$  macros in that file.



### 3.2 Typesetting T<sub>E</sub>X and L<sup>A</sup>T<sub>E</sub>X Codes

The L<sup>A</sup>T<sub>E</sub>X codes in this manual are typeset using the `bidicode` package. In standard L<sup>A</sup>T<sub>E</sub>X you can not use footnotes inside `\chapter`, `\part`, `\section`, `\subsection`, `\subsubsection` and any other section-like commands, `\caption` and `tabular` environment. `bidi` package provides `bidiftnxtra` package that solves the issue of footnote in standard L<sup>A</sup>T<sub>E</sub>X. `bidiftnxtra` package should be loaded after `bidi` package.

### 3.3 Typesetting Poems

The `bidi` package provides `bidipoem` package for typesetting Persian poems. It provides four environments, `traditionalpoem`, `modernpoem` and starred version of these. In the starred version of these environments you do not need to type `\\` and that is the only difference with the normal version of the environments. The `traditionalpoem` environment and its starred version are also useful for typesetting Classic Arabic poetry, in fact this package may also be useful for other RTL languages.

When using `bidipoem` package, at least you need to run `xelatex` twice on your document. In fact, if you run `xelatex` just once on your document, you get a message saying “Unjustified poem. Rerun XeLaTeX to get poem right”.

When you typeset your poems, you might get underfull `\hbox` messages. This is absolutely normal and if you want to get rid of these underfull `\hbox` messages, then you would need to use `Kashida`.

If you need to change the default distance between two verses, you can do just that by:

```
\renewcommand\poemcolsepskip{<length>}
```

```
\begin{traditionalpoem}
<verse1>&<verse2>\\
<verse3>&<verse4>\\
...
\end{traditionalpoem}
```

```
\begin{traditionalpoem*}
<verse1>&<verse2>
<verse3>&<verse4>
...
\end{traditionalpoem*}
```

### 3.4 Typesetting Résumé

The `bidi` package provides `bidimoderncv`<sup>2</sup> class for typesetting resumé. There are two examples, namely `test-casualcv.tex` and `test-classiccv.tex`, in the `doc` folder than you can look and learn how you can use it.

### 3.5 Print Two Pages On A Single Page

`bidi` package provides `bidizin1` package for printing two pages on a single (landscape) A4 page. Page numbers appear on the included pages, and not on the landscape ‘container’ page.

<sup>2</sup>This class is the modified version of `moderncv` class.



### 3.6 Producing Presentations

At the moment, there is only one class that you can prepare your presentations with.

## 4 Some Useful Internal Macros and Programming Tips

There are some useful internal macros and programming tips that might be helpful for you. This section, explains all these useful internals and programming tips.

### 4.1 Equating Conditionals

`\eqnewif{<\newconditional1>}{<\newconditional2>}`

In standard  $\TeX$ , `\newif` command is provided that you can define a new conditional with it. `\eqnewif` command is similar to `\newif` command but:

- ☞ With `\eqnewif` command, you can define two new conditionals instead one, so clearly it has two mandatory arguments.
- ☞ `\newconditional1` will be identical to `\newconditional2`, so that whenever `\newconditional1` is true, then `\newconditional2` is also true and whenever `\newconditional1` is false, then `\newconditional2` is also false and vice versa.

### 4.2 RTL Conditional

`\if@RTL`

`\if@RTL` conditional is true inside RTL mode and it is false in LTR mode.

### 4.3 Main RTL Conditional

`\if@RTLmain`

If the main direction of the document is RTL, `\if@RTLmain` is true and if the main direction of the document is LTR, `\if@RTLmain` is false.

### 4.4 Latin Conditional

`\if@nonlatin`

The packages should set the value of `\if@nonlatin` inside any environment that uses LTR script to false (`\@nonlatinfalse`) and inside any environment that uses RTL script to true (`\@nonlatintrue`).

### 4.5 Tags Internal Macro

`\@iftagsloaded{<tags name>}{<do thing(s) if the tag is loaded>}`  
`{<do thing(s) if the tag is not loaded>}`

As you can see, the syntax of `\@iftagsloaded` is exactly the same as the syntax of `\@ifpackageloaded` and `\@ifclassloaded`. By tags, we mean things like `leqno` or



reqno. Please note that in the argument  $\langle tags\ name \rangle$ , the extension `clo` should not be given.

## 4.6 Definition File Loaded Internal Macro

```
\@ifdefinitionfileloaded{ $\langle definition\ file\ name \rangle$ }
{ $\langle do\ thing(s)\ if\ the\ definition\ file\ is\ loaded \rangle$ }
{ $\langle do\ thing(s)\ if\ the\ definition\ file\ is\ not\ loaded \rangle$ }
```

As you can see, the syntax of `\@ifdefinitionfileloaded` is exactly the same as the syntax of `\@ifpackageloaded` and `\@ifclassloaded`. By definition file, we mean things like `hyperref-bidi.def` or `wrapfig-bidi.def`. Please note that in the argument  $\langle definition\ file\ name \rangle$ , the extension `def` should not be given.

## 4.7 Tabular Conditional

```
\if@RTLtab
```

If the tabular is typeset RTL, `\if@RTLtab` is true and if the tabular is typeset LTR, `\if@RTLtab` is false.

## 4.8 Footnote Conditional

```
\if@RTL@footnote
```

When footnotes are typeset RTL, `\if@RTL@footnote` is true and when footnotes are typeset LTR, `\if@RTL@footnote` is false.

## 4.9 Direction Ensuring Macros

```
\@ensure@RTL{ $\langle text \rangle$ } \@ensure@RL{ $\langle text \rangle$ } \@ensure@LTR{ $\langle text \rangle$ }
\@ensure@LR{ $\langle text \rangle$ } \@ensure@dir{ $\langle text \rangle$ } \@ensure@maindir{ $\langle text \rangle$ }
```

- ☞ `\@ensure@RTL` and `\@ensure@RL` internals make sure that  $\langle text \rangle$  is always typeset RTL, independent on the current mode.
- ☞ `\@ensure@LTR` and `\@ensure@LR` internals make sure that  $\langle text \rangle$  is always typeset LTR, independent on the current mode.
- ☞ `\@ensure@dir` and `\@ensure@maindir` if used in RTL mode, they put  $\langle text \rangle$  inside `\RLE` and if used in LTR mode, they put the text as it is.

## 4.10 Reset Direction Macro

```
\save@dir \saved@@dir \reset@dir
```

- ☞ `\save@dir`, if the direction of typesetting is RTL, defines `\saved@@dir` to be RTL and if the direction of typesetting is LTR, defines `\saved@@dir` to be LTR.
- ☞ `\reset@dir`, if `\saved@@dir` is defined as RTL, inserts `\setRTL` otherwise, if `\saved@@dir` is defined as LTR, inserts `\setLTR`, otherwise does nothing.



## Part II

# Plain T<sub>E</sub>X Manual

## 5 Basics

### 5.1 Loading The Package

You can load the package in the ordinary way;

```
\input bidi
```

When loading the package, it is important to know that: `bidi` should be the last package that you load, because otherwise you are certainly going to overwrite `bidi`'s definitions and consequently, you will not get the expected output.

### 5.2 Commands for Version number, and Date of The Package

```
\bidiversion \bididate
```

☞ `\bidiversion` gives the current version of the package.

☞ `\bididate` gives the current date of the package.

```
1 \input bidi
2 This is typeset by \textsf{bidi} package, \bidiversion, \bididate.
3 \end
```

### 5.3 Turning TeX--XeT features on and off

The bidirectional typesetting in XeT<sub>E</sub>X is based on TeX--XeT algorithm and `bidi` package automatically turns the feature on for typesetting bidirectional texts. As the internal data structures built by TeX--XeT differ from those built by T<sub>E</sub>X, the typesetting of a document by TeX--XeT may therefore differ from that performed by T<sub>E</sub>X. So you may notice that some packages behave differently when TeX--XeT feature is on and you may want to turn the feature off for a part of the text so that you get the default behaviour of original T<sub>E</sub>X. Two commands are provided for this purpose:

```
\TeXXeTOn \TeXXeTOff
```

☞ `\TeXXeTOn` turns TeX--XeT feature on, which is active by default when `bidi` package is loaded.

☞ `\TeXXeTOff` turns TeX--XeT feature off.



## 5.4 Paragraph Switching Commands

<code>\setLTR</code>	<code>\setLR</code>	<code>\unsetRL</code>	<code>\unsetRTL</code>
<code>\setRTL</code>	<code>\setRL</code>	<code>\unsetLTR</code>	

☞ With any of the commands in the first row, you can typeset LTR paragraphs.

☞ With any of the commands in the second row, you can typeset RTL paragraphs.

```
1 \input bidi
2 \setRTL%
3 Anyone who reads Old and Middle English literary texts will be
4 familiar with the mid-brown volumes of the EETS, with the symbol
5 of Alfred's jewel embossed on the front cover.
6
7 \setLTR% Notice the blank line before \setLTR
8 Anyone who reads Old and Middle English literary texts will be
9 familiar with the mid-brown volumes of the EETS, with the symbol
10 of Alfred's jewel embossed on the front cover.
11 \end
```

## 5.5 Paragraph Switching Environments

<code>\LTR</code>	<code>&lt;text&gt;</code>	<code>\endLTR</code>
<code>\RTL</code>	<code>&lt;text&gt;</code>	<code>\endRTL</code>

☞ With LTR environment, you can typeset LTR paragraphs.

☞ With RTL environment, you can typeset RTL paragraphs.

```
1 \input bidi
2 \RTL
3 Anyone who reads Old and Middle English literary texts will be familiar
4 with the mid-brown volumes of the EETS, with the symbol
5 of Alfred's jewel embossed on the front cover.
6 \LTR
7 Anyone who reads Old and Middle English literary texts will be familiar
8 with the mid-brown volumes of the EETS, with the symbol
9 of Alfred's jewel embossed on the front cover.
10 \endLTR
11 And we are still typesetting right to left.
12 \endRTL
13 \end
```

## 5.6 Typesetting Short LTR and RTL Texts

<code>\LRE{&lt;text&gt;}</code>	<code>\LR{&lt;text&gt;}</code>
<code>\RLE{&lt;text&gt;}</code>	<code>\RL{&lt;text&gt;}</code>

☞ With any of the commands in the first row, you can typeset short LTR text inside RTL paragraphs.



☞ With any of the commands in the second row, you can typeset short RTL text inside LTR paragraphs.

```

1 \input bidi
2 \RTL
3 Anyone who reads Old and Middle English \LRE{Short left to right text}
4 literary texts will be familiar
5 with the mid-brown volumes of the EETS, with the symbol
6 of Alfred's jewel embossed on the front cover.
7 \LTR
8 Anyone who reads Old and Middle English \RLE{Short right to left text}
9 literary texts will be familiar
10 with the mid-brown volumes of the EETS, with the symbol
11 of Alfred's jewel embossed on the front cover.
12 \endLTR
13 \endRTL
14 \end

```

## 5.7 Primitive-like commands

```
\hboxR \hboxL \vboxR \vboxL
```

- ☞ The syntax of `\hboxR` is exactly the same as the syntax of `\hbox`, but its contents is always typeset RTL.
- ☞ The syntax of `\hboxL` is exactly the same as the syntax of `\hbox`, but its contents is always typeset LTR.
- ☞ The syntax of `\vboxR` is exactly the same as the syntax of `\vbox`, but its contents is always typeset RTL.
- ☞ The syntax of `\vboxL` is exactly the same as the syntax of `\vbox`, but its contents is always typeset LTR.

## 5.8 Something To know about `\hbox`

If you enable RTL typesetting and typeset an horizontal box at the beginning of the document:

```

1 \input bidi
2 \setRTL
3 \hbox{This is a Test}
4 \end

```

You see that even you have used `\setRTL`, the horizontal box appears LTR (It appears on the left hand side and its content is typeset left to right). This is because when `TEX` starts, it is in the vertical mode so if you need to have that `\hbox` appear RTL, then write `\leavevmode` before `\hbox`:

```

1 \input bidi
2 \setRTL
3 \leavevmode\hbox{This is a Test}
4 \end

```



## 5.9 Typesetting Logos

`\XeTeX`

`bidi`  defines XeTeX logo and in addition, it makes sure that the logo, TeX is typeset LTR.

## 6 Some Useful Internal Macros and Programming Tips

There are some useful internal macros and programming tips that might be helpful for you. This section, explains all these useful internals and programming tips.

### 6.1 RTL Conditional

`\if@RTL`

`\if@RTL` conditional is true inside RTL mode and it is false in LTR mode.

### 6.2 Main RTL Conditional

`\if@RTLmain`

If the main direction of the document is RTL, `\if@RTLmain` is true and if the main direction of the document is LTR, `\if@RTLmain` is false.

### 6.3 Direction Ensuring Macros

`\@ensure@RTL{<text>}`   `\@ensure@RL{<text>}`   `\@ensure@LTR{<text>}`  
`\@ensure@LR{<text>}`   `\@ensure@dir{<text>}`   `\@ensure@maindir{<text>}`

- ☞ `\@ensure@RTL` and `\@ensure@RL` internals make sure that `<text>` is always typeset RTL, independent on the current mode.
- ☞ `\@ensure@LTR` and `\@ensure@LR` internals make sure that `<text>` is always typeset LTR, independent on the current mode.
- ☞ `\@ensure@dir` and `\@ensure@maindir` if used in RTL mode, they put `<text>` inside `\RLE` and if used in LTR mode, they put the text as it is.

### 6.4 Reset Direction Macro

`\save@dir`   `\saved@@dir`   `\reset@dir`

- ☞ `\save@dir`, if the direction of typesetting is RTL, defines `\saved@@dir` to be RTL and if the direction of typesetting is LTR, defines `\saved@@dir` to be LTR.
- ☞ `\reset@dir`, if `\saved@@dir` is defined as RTL, inserts `\setRTL` otherwise, if `\saved@@dir` is defined as LTR, inserts `\setLTR`, otherwise does nothing.

