

# **L<sup>A</sup>T<sub>E</sub>X2RTF**

---

A converter from L<sup>A</sup>T<sub>E</sub>X to RTF

Edition 0.7.4

for program version 1.9.19

**Georg Lehner,**

updated by Wilfried Hennings and Scott Prahl, with contributions by Mikhail Polianski

---

Copyright © 1998-2007 Georg Lehner, Wilfried Hennings, Scott Prahl

Permission is granted to make and distribute verbatim copies of this manual provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this manual under the conditions for verbatim copying, provided also that the sections entitled “Copying” and “GNU General Public License” are included exactly as in the original, and provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this manual into another language, under the above conditions for modified versions, except that this permission notice may be stated in a translation approved by the Free Software Foundation.

# 1 Introduction

`LATEX2RTF` is a translator program from `LATEX` text into “rich text format” files. These files are commonly referred to as RTF files. RTF is a published standard format by Microsoft. This standard can be ambiguous in places and Microsoft ignores parts of the standard it finds inconvenient, but RTF is widely used by many WYSIWIG text editors and is supported by Microsoft Word and most text processors.

`LATEX2RTF` translates the text and as much of the formatting information from `LATEX` to RTF. Be forewarned that the typeset output is not nearly as good as what you would get from using `LATEX` directly. So, why bother translating? Consider,

1. You use `LATEX` and hate everything beginning with MS-... Nevertheless, you have to share your documents with people who don't even know that there are alternatives to MS-...
2. You know somebody who frequently sends you very fine `LATEX` documents. Unfortunately, you are “on the other side” and need to import her files, steal some part, and then desktop publish it in your fine MS-... environment.
3. You like `LATEX` and `BibTEX`. You interact with the rest of the world. You know someone that wants to include your writing in a Word document.

There are drawbacks to the conversion process. In fact, don't expect any `LATEX` file to be converted as you would like, don't expect it to be converted without errors or warnings, and don't be especially surprised when it doesn't convert at all. `LATEX2RTF` is known to have many bugs and many missing features. Paradoxically, this number seems to grow more and more with each day. However, we can categorically state that there are some special cases in which a `LATEX` file will be translated to RTF satisfactorily by `LATEX2RTF` — This was sort of disclaimer, ok? OK!

`LATEX` is a system for typesetting text and therefore it focuses on the logical structure of a document, whilst RTF is meant to be a transport format for a family of Desktop Publishing Software, dealing mostly with the design of a text.

Although the commands and styles in `LATEX` are much more flexible and standardized than in RTF, only a small subset of commands has been implemented to date (see [Section 8.1 \[Unimplemented Features\]](#), page 29).

Some of the capabilities of `LATEX2RTF` are restricted in scope or buggy (see [Section 8.3 \[Known Bugs\]](#), page 29).

RTF is a moving target, because Microsoft does not stop inventing new extensions and features; consequently you cannot view newer RTF files with older word processors. The syntax and semantics of RTF are somewhat artistic, i.e., you can generate a syntactically correct RTF file that cannot be displayed by some/most word processors. For more details on RTF the specification consult the links at <http://latex2rtf.sf.net/>



## 2 Installation

### 2.1 General

The documentation of the program is found in the ‘doc/’ directory in the file ‘`latex2rtf.texi`’ in the GNU TeXInfo format. For your convenience, you can find HTML and PDF versions of the manual there as well.

### 2.2 Obtaining L<sup>A</sup>T<sub>E</sub>X2RTF

L<sup>A</sup>T<sub>E</sub>X2RTF is available for many Unix Platforms, for the Macintosh, and for MS-DOS, including all versions of MS Windows.

The latest version of L<sup>A</sup>T<sub>E</sub>X2RTF is available at [SourceForge](http://www.dante.de) and — with some delay — on CTAN sites: e.g., <http://www.dante.de> or <http://www.ctan.org>.

The DOS package (1.8aa and up) will also run under all MS Windows versions. It requires an i386 processor or better.

It should only be used on older Microsoft systems (before Win95) because it has known issues when converting more than 26 equations to bitmaps.

For Win9x, ME, NT, or newer, you should use the win or win-NO-GUI package.

The win-NO-GUI package (starting from L<sup>A</sup>T<sub>E</sub>X2RTF 1.9.19) is identical to the DOS package except that it is compiled with Cygwin (up to 1.9.19) or MinGW (1.9.20 and up), and only runs under Win32 (Win9x, ME, NT, 2000, XP, 2003, Vista).

The win package (starting from L<sup>A</sup>T<sub>E</sub>X2RTF 1.9.15) is the DOS package (up to L<sup>A</sup>T<sub>E</sub>X2RTF 1.9.17) or the win-NO-GUI package (L<sup>A</sup>T<sub>E</sub>X2RTF 1.9.18 and up) plus Win32 GUI program (l2rshell).

Up to version 1.9.19 it does not run under 64bit systems. From version 1.9.20 it also runs under 64bit systems.

The win64 package (L<sup>A</sup>T<sub>E</sub>X2RTF 1.9.19) is the win package but with 32bit versions of the shell utilities so that it also runs on 64bit systems.

There are a few people working on L<sup>A</sup>T<sub>E</sub>X2RTF, and some more make contributions, coordinated by Wilfried Hennings (texconvfaq “at” gmx.de). See the [SourceForge](http://www.dante.de) project pages for the latest news.

### 2.3 UNIX

To install,

1. Edit ‘Makefile’ for your local configuration. In particular, pay attention to the `PREFIX` variable. If you do not have root access you might wish to set the makefile variable `PREFIX` to be your home directory.

On some machines the cc compiler will issue errors. Therefore the default compiler command in the Makefile is `CC=gcc`.

2. As of version 1.9.13, L<sup>A</sup>T<sub>E</sub>X2RTF supports conversion of L<sup>A</sup>T<sub>E</sub>X equations to bitmaps using the shell script ‘`latex2png`’, found in ‘`scripts/`’. ‘`latex2png`’ requires that both L<sup>A</sup>T<sub>E</sub>X and ‘`ImageMagick`’ are installed. L<sup>A</sup>T<sub>E</sub>X2RTF will translate documents without a working ‘`latex2png`’, but some features will be missing. You can verify that the ‘`latex2png`’ script is working by typing `make` in the ‘`scripts/`’ directory.

### 3. make

If this is not your first time installation, you may want to preserve your old configuration ('\*.cfg') files. Copy them to a safe place before installing.

On IBM AIX, the IBM make utility does not support some of the commands used in Makefile. In this case use `gmake` (from GNU) instead.

Sun has decided to support the XPG4 standard on Solaris by an alternative set of binaries. To allow bitmap conversion of equations, two things are needed. First, change the first line of `latex2png` to `#!/usr/xpg4/bin/sh`. Second, define the `XPG4_GREP` environment variable accordingly, for bash-like shells `XPG4_GREP=/usr/xpg4/bin/grep`; `export XPG4_GREP` or for tsch-like shells `setenv XPG4_GREP /usr/xpg4/bin/grep`.

### 4. make install

If your `mkdir` doesn't support the `-p` option, then create the necessary directories by hand and remove the option from the `$MKDIR` variable. If you have other problems, just copy '`latex2rtf`' and '`latex2png`' to a binary directory, and move the contents of the '`cfg/`' directory to the location specified by `$CFG_INSTALL`.

### 5. make check

[OPTIONAL] This tests `LATEX2RTF` on a variety of `LATEX` files. Expect a whole lot of warnings, but no outright errors. (On IBM AIX, use `gmake check`.) Note that this will check the basic functionality of the '`latex2png`' script, and then that of '`latex2rtf`'.

### 6. make install-info

[OPTIONAL] This installs '`.info`' files for use with the `info` program.

You no longer need to define the environment variable `RTFPATH`. This is only necessary if you move the directory containing the '`.cfg`' files. Just define `RTFPATH` to be the path for the new location of the '`cfg`' directory.

## 2.4 DOS

The UNIX and Mac packages do not contain an executable for DOS or Windows. You may compile it yourself (with `djgpp` or `MinGW`) or get the DOS package as file '`latex2rtf-x.x.x_dos.zip`' (where `x.x.x` is the version number) or the Windows package as file '`latex2rtf-x.x.x_win.zip`' from [SourceForge](#)

The DOS distribution contains a precompiled executable which should run under plain DOS and also in the command prompt (or "console") of any MS Windows system including Vista.

**Limitations of the DOS version:** The DOS version has a bug which causes conversion of equations to bitmaps to fail for the remaining equations after successfully converting the first 26 of them.

**So if you run a Win32 system** (i.e. Win 9x, ME, NT, 2000, XP, 2003, Vista) **it is highly recommended to use the Windows package.**

To install the DOS or Windows package of `LATEX2RTF`, extract all files from the zip archive, preserving the folder structure (winzip: check "use folder names"), preferably to '`C:\l2r`', or under Windows, to your "Program Files" folder (which ever it is named in your system) because then it will find its `cfg` files by default.

If you extracted them to another folder (e.g. ‘C:\my files\l2r’), either edit the file ‘L2RPREP.BAT’ and change the folder ‘C:\l2r’ to where you put them, or specify the path to the cfg folder in the command line.

Make sure that the folder containing the file ‘L2RPREP.BAT’ is in your search path, or put this file into a folder which is in your search path.

To display the current search path, enter ‘PATH’ from the command prompt (with no arguments).

Under plain DOS, conversion of equations or figures to bitmaps is not possible because Ghostscript and ImageMagick are not available for plain DOS.

## 2.5 Win32 systems

To install  $\text{\LaTeX}2\text{R}$  on a Win32 system (WinNT, Win2000, WinXP, WinVista), download the win-gui package, execute the ‘`latex2rtf-x.x.x_win.exe`’ (where x.x.x is the version number) and follow the instructions.

Note: The installer and GUI shell do not support Win9x and WinMe since  $\text{\LaTeX}2\text{R}$  v.1.9.17. If you have one of these systems, you should use the win package (1.9.19 and up) of  $\text{\LaTeX}2\text{R}$ .

To start the program double-click the  $\text{\LaTeX}2\text{R}$  icon, or drag and drop a ‘.tex’ file onto the icon.

If your  $\text{\LaTeX}$  document refers to external graphic files, e.g. eps, or you want to use the option to convert equations to bitmaps, you must have LaTeX, ImageMagick and Ghostscript installed. These programs are freely available for download at <http://www.miktex.org/>, <http://www.imagemagick.org> and <http://www.ghostscript.com>

If MikTeX, ImageMagick and Ghostscript are installed, the paths to their executables are detected automatically. If for any reason this doesn’t happen, these paths can be specified manually in the “Environment” tab of the  $\text{\LaTeX}2\text{R}$  GUI shell window.

If you need to start  $\text{\LaTeX}2\text{R}$  not from its own shell but from another Windows application, you need to add the folder containing the latex2png script and the helper programs to the search path.

Some applications (e.g. Lyx) allow to temporarily add the path by specifying it in the options.

If this is not possible, you can add the latex2rtf folder to Windows’ default path like follows:

To add a folder – e.g. C:\Program Files\latex2rtf – to the search path:

- Win95, Win98, WinME
  - Open the file ‘C:\Autoexec.bat’ either with Edit (DOS) or with Notepad (Windows).
  - At the end of that file, add the line  
‘PATH=C:\PROGRA~1\latex2rtf;%PATH%’
  - Save the file to its original location. Shutdown and reboot the PC.
- WinNT, Win2000, WinXP
  - Right-click “My Computer” (German: “Arbeitsplatz”), then select “Properties”;

- NT: Click the “Environment” tab;  
XP: Click the “Extended” tab, then the “Environment variables” button;
- Find the line beginning with ‘PATH=’ and insert the string  
‘C:\PROGRA~1\latex2rtf;’  
so that the complete line looks like  
‘PATH=C:\PROGRA~1\latex2rtf;C:\WINDOWS;...’
- Click "OK"

Either the folders where TeX, ImageMagick and Ghostscript are installed must be in your search path, or you must edit the file ‘L2RPREP.BAT’, ensure that the pathes in this file point to the folders where TeX, ImageMagick and Ghostscript are installed on your machine, and call ‘L2RPREP’ before calling ‘latex2rtf’.

Or you edit ‘L2R.BAT’ and call ‘L2R’ instead of calling ‘LATEX2RT.EXE’.

When equations are to be converted to bitmaps, for each equation  $\text{\LaTeX2RTF}$  first writes a temporary .tex file to disk which consists of only the equation to be converted. It then sends the call for “`bash latex2png l2r_nnnn.tex`” to the operating system.

In previous versions of latex2rtf, some users got the message “Out of environment space” (can occur only under Windows 95, Windows 98 or Windows ME); this should be solved now by using bash.exe instead of command.com.

Some notes on Windows Vista compatibility:

I only tested  $\text{\LaTeX2RTF}$  under Vista with the following system settings:

- User Account Control OFF
- Install and run all programs with admin rights.

With these settings, the Windows installer of  $\text{\LaTeX2RTF}$  and the following helper software versions were successfully installed and used:

- MikTeX 2.6
- Ghostscript 8.60
- ImageMagick-6.3.6-10-Q8-windows-dll.exe

## 2.6 Macintosh

If you want a MacOS X version, make sure that you have installed the developer tools CD that is appropriate for your OS version, and then follow the directions above for a UNIX installation. Alternatively you can install using fink <http://fink.sourceforge.net/> or as an i-installer package <http://www.rna.nl/ii.html>.

As of 2006-01-30 there also is a GUI shell for Macintosh.

See <http://www.inf.ethz.ch/personal/fischerk/LaTeX2rtf/index.html>

There is a PPC port of an old version 1.9k for Classic MacOS  $\text{\LaTeX2RTF}$ . To convert a  $\text{\LaTeX}$  file using this version, drag the file onto the  $\text{\LaTeX2RTF}$  icon. The translation is best if there are ‘.aux’ and ‘.bbl’ files in the same folder as the ‘.tex’ file to be converted. These should be generated using  $\text{\LaTeX}$  and ‘bibtex’.

## 2.7 Problems Compiling

The code for  $\text{\LaTeX2RTF}$  is standard ANSI C. Some possible pitfalls are

- Not correctly defining your compiler in the Makefile. The default is to use gcc.



- Encountering errors because the compiler options. During development all compiler warnings are turned on. However, different compilers have different interpretations of ‘-Wall’ and may generate errors that were not found in a different development system. Please report these, but a quick fix is to remove all compiler options.

## 2.8 Problems with make check

All the files in the ‘test’ directory are converted (with varying degrees of success) using `LATEX2RTF` and are tested before most CVS check-ins and with all released tarballs. There will be many warning messages, but there should be no actual error messages. If you do not have a working latex2png script, then some of the files will fail to be translated.



## 3 Using LaTeX2RTF

### 3.1 General Assumptions

LaTeX2RTF assumes that the ‘.tex’ file you want to convert is a valid LaTeX document. The chances of a successful LaTeX2RTF conversion are slightly better than the proverbial snowball’s if the ‘.tex’ file doesn’t latex properly. Use LaTeX to find and correct errors before using LaTeX2RTF.

To correctly convert font names you must edit the ‘fonts.cfg’ configuration file. This file is used to specify the needed font names and how the LaTeX default font names should be converted to RTF (see [Section 5.6 \[Font Configuration\]](#), page 23). LaTeX variables and user defined commands are not evaluated. They will be simply ignored. To let LaTeX2RTF know the names of variables you can add them in the ‘ignore.cfg’ file (see [Section 5.5 \[Ignore Command\]](#), page 22).

The environment variable RTFPATH may contain a search path for the support files (all files ending in ‘.cfg’). If no file is found during the search in the search-path or if the environment variable is not set, the compiled-in default for the configuration-file directory is used. If the files are not found at all the program aborts.

In the MS-DOS version the search path is separated by ‘;’ in the Unix version by ‘.’. For the paths themselves apply ‘\’ and ‘/’. A separator may appear at the beginning or ending of RTFPATH.

Make sure that the configuration files are in the correct directory. LaTeX2RTF will need at least ‘fonts.cfg’, ‘direct.cfg’, ‘ignore.cfg’, ‘english.cfg’. You may have to change one or more of them to suit your needs (see [Chapter 5 \[Configuration\]](#), page 21).

See [Section 8.2 \[Missing options\]](#), page 29, for actual implementations irregularities.

See [Section 8.4 \[Reporting Bugs\]](#), page 29, for information on how to reach the maintainer.

### 3.2 LaTeX2RTF Options

The LaTeX2RTF command converts a LaTeX file into RTF text format. The text and much of the formatting information is translated to RTF making the new file look similar to the original. The command line syntax is:

```
latex2rtf [-options] inputfile[.tex]
```

or for the DOS and Windows versions:

```
latex2rt [-options] inputfile[.tex]
```

The -options may consist of one or more of the following

**-a auxfile**

specify an ‘.aux’ file (for table and figure references) that differs from ‘inputfile.aux’. If this is omitted, the name of the inputfile with the suffix replaced ‘.aux’ will be taken. You must provide both files (‘.tex’ and the ‘.aux’) to be able to convert cross-references in a LaTeX file. The ‘.aux’ is created by running the ‘inputfile.tex’ through latex.

**-b bblfile**

Unless an ‘bblfile’ is specified with the **-b** option, `LaTeX2RTF` uses a ‘inputfile.bbl’. The ‘bblfile’ file is used for citations and is typically created by running ‘inputfile.aux’ through ‘bibtex’.

**-C codepage**

used to specify the character set (code page) used in the `LaTeX` document. This is only important when non-ansi characters are included in the `LaTeX` document. Typically this is done in a `LaTeX2ε` file by using `\usepackage[codepage]{inputenc}` and in this case you need not specify the **-C** codepage option. If NO `\usepackage[codepage]{inputenc}` is in the `LaTeX2ε` file, you must inform the converter about the codepage by the **-C** codepage option. You may select any of the following code pages: ansinew, applemac, cp437, cp437de, cp850, cp852, cp865, decmulti, cp1250, cp1252, latin1, latin2, latin3, latin4, latin5, latin9, next. The default behavior is to use ansinew (code page 1252). Cyrillic support includes conversion of koi8-r, koi8-u, cp1251, cp855, cp866, maccyr, and macukr encodings.

**-d debug\_level**

The ‘-d’ option determines the amount of debugging information to send to stderr while translating. `debug_level=0` means only Errors, ‘1’ Warning Messages (default) also. The `debug_level` can go as high as ‘7’ for insane amounts of debugging fun.

**-D dots\_per\_inch**

used to specify the number of dots per inch for equations converted to bitmaps. This value is also used when picture environments are converted to bitmaps as well as when EPS graphics are converted to png files. The default value is 300 dots per inch.

**-f#**

where # selects which fields to use during conversion:

- f0** do not use fields in RTF. This is handy when primitive RTF editors are being used to view the RTF output.
- f1** use fields for equations but not `\ref` and `\cite`.
- f2** use fields for `\ref` and `\cite` but not equations. This will be useful for versions of OpenOffice that import cross-references properly (as of Sept 2003 in a soon-to-be released version) but do not properly handle fields in equations.
- f3** use fields when possible. This is the default and is most useful when the RTF file is being exported to be used in Word. This retains the most information from the original `LaTeX` file.

**-F**

use LaTeX to create bitmaps for all figures. This may help when figures are not translated properly with the default settings. This typically requires a functional version of ImageMagick on your machine to work properly.

**-h**

a short usage description

**-i language**

used to set the idiom or language used by the  $\text{\LaTeX}$  document. Typically, this is specified in a  $\text{\LaTeX} 2_{\epsilon}$  document by including `\usepackage[language]{babel}` where `language` is one of the languages supported by the `babel` package. All languages listed in the `babel` system are supported so far as translations for “Chapter,” “References,” and the like. Furthermore, some commands found in the style files for german, french, russian, and czech style are supported (see [Section 5.7 \[Language Configuration\]](#), page 23).

**-l** same as ‘`-i latin1`’ (Note that the default behavior is to use ‘`ansinew`’ which is a superset of ‘`latin1`’). Included for backwards compatibility.

**-M#** where `#` selects the type of equation conversion. Use

- M1** convert displayed equations to RTF
- M2** convert inline equations to RTF
- M4** convert displayed equations to bitmap
- M8** convert inline equations to bitmap
- M16** insert Word comment field that contains the original equation text
- M32** insert the raw latex equation delimited by `<<:` and `:>>`. This is potential useful when using version 6 of the MathType equation editor, which converts typed or pasted TeX code into a MathType equation.  
Probably this could also be useful for use in OpenOffice, as OO has an equation syntax which partially resembles TeX syntax.

These switches can be combined to get different effects. Handy examples are

- M3** convert both inline and displayed equations to RTF (default)
- M6** convert inline equations to RTF and displayed equations to bitmaps
- M12** convert both inline and displayed equations to bitmaps

Bitmap conversion requires that you have installed a working `latex2png` script. Producing bitmaps is slow.

When running the DOS version, conversion to bitmaps works for the first 26 equations but fails for the rest with the message

`"latex2png: pipe error: Too many open files (EMFILE)".`

This is probably a bug in the `djgpp 2.04` compiler; I chose this version because of its better support of long filenames under Win32 (95, 98, ME, 2000, XP, Vista).

The Windows version, compiled with Cygwin in MinGW mode, successfully converts at least 79 equations to bitmaps.

**-o outputfile**

Unless an ‘`outputfile`’ is specified with the `-o` option, the resulting RTF filename is formed by removing ‘`.tex`’ from the ‘`inputfile`’ and appending ‘`.rtf`’.

**-p** Escape parentheses in mathematical formulas. This has no effect unless EQ fields are being generated. When this option is used, then in a ‘(’ or ‘)’ that appears in an EQ field will be preceded by a backslash. Despite documentation to the contrary (which says that all parentheses should be escaped), adding escapes usually produces a worse result than doing nothing. If Word displays some formulas with parentheses as “Error!”, you might try this option as a last resort. See also the **-S** option.

This is an option because it will break typesetting equations with non-matching parentheses (because an unmatched unquoted parenthesis would terminate the field).

**-P /path/to/cfg**

used to specify the folder (i.e. directory) that contains the `.cfg` files and/or the folder that contains the `latex2png` script.

Unix, Mac: The folder that contains the `latex2png` script must be prepended by a `":"`.

DOS, Windows: The scripts folder is not used, the scripts are always taken from the search path, therefore the folder containing the scripts and the helper programs must be the first in the search path. You can either add it to the search path permanently or add it temporarily by calling `l2rprep.bat` before starting conversions.

If any of the folder names contains a blank, the folder string must be enclosed in single (Unix, Mac) or double (DOS, Windows) quotes.

Examples:

(Unix:) `latex2rtf -P ./cfg/./scripts/ foo`

(DOS/Windows:) `latex2rt -P "C:\Program Files\latex2rtf\cfg" foo`

Note that without specifying the **-P** option, `LATEX2RTF` tries to find its `cfg` files in the following locations:

1. the folder specified by the environment variable `RTFPATH`, if this variable exists;
2. the folder `C:%PROGRAMFILES%/latex2rtf/cfg`, if the variable `PROGRAMFILES` exists – this is the folder in which `LATEX2RTF` is installed by the Windows GUI installer with default settings;
3. the folder specified at compilation time by the variable `CFGDIR`. This is set in the Makefile. The DOS and Windows versions are compiled with `CFGDIR=C:/l2r`.

**-se#** selects the scale for equation conversion, where `#` is the scale factor (default 1.22).

**-sf#** selects the scale for figure conversion, where `#` is the scale factor (default 1.35).

**-t#** where `#` selects the type of table conversion. Use

**-t1** convert tables to RTF (default)

**-t2** convert tables to bitmaps

**-v or -V** prints version information on standard output and exits.

- S            used to specify that semicolons should be used to separate arguments in RTF fields (instead of commas). Typically this is needed when the machine that opens the RTF file has a version of Windows that uses ‘,’ for decimal points.
- T /path/to/tmp            used to specify the folder where to put temporary files. Not used by the DOS and Windows versions.
- W            includes warnings directly in the RTF file
- Z#            add the specified number of extra } to the end of the RTF file. This is useful for files that are not cleanly converted by LaTeX2RTF .

With no arguments other than switches starting with a “-”, LaTeX2RTF acts as a filter, i.e., it reads from `stdin` and writes to `stdout`. In addition, diagnostic messages are sent to `stderr`. If these standard channels are not redirected using `<` and `>`, then the input is read from the command line, and both output and error messages are printed on the screen.

If a non-switch argument is present, LaTeX2RTF assumes it is the name of the input file. The file must have extension “.tex” but the extension is optional. The output file is constructed from the input file name by removing the extension “.tex” and adding “.rtf”.

### 3.3 Debugging

With the ‘-d’ option you can specify how much processing information LaTeX2RTF reports. If there is a logfile specified the output goes to this file. Nonetheless Warnings and Errors are logged to `stderr` always.

Possible values of ‘-d’ are

0. only errors.
1. Translation Warnings (default).
2. shows preparsing of sections
3. Reasonably high level debugging messages
4. Show all function calls
5. Show each character as it is processed
6. Show processing of characters as they are output as well





## 4 Features

In this chapter you find what styles  $\text{\LaTeX2RTF}$  is supposed to translate correctly to RTF.

### 4.1 LaTeX2e

$\text{\LaTeX2RTF}$  understands most of the commands introduced with  $\text{\LaTeX2}_\epsilon$ . It translates both the old 2.09 version of `\documentstyle[options]{format#}` and the newer `\documentclass[options]{format}`.

### 4.2 Unicode Support

As of version 1.9.17,  $\text{\LaTeX2RTF}$  has limited unicode support.  $\text{\LaTeX}$  files that use unicode sequences are just emitted directly into the RTF file. Symbols and odd characters in math sequences may also be converted to a unicode sequence if there is no corresponding symbol in the “Symbol” or “MT Extra” fonts.

Support for unicode encoded input is activated by the  $\text{\LaTeX}$  command  
`\usepackage[utf8]{inputenc}`  
 or  
`\usepackage[utf8x]{inputenc}`.

### 4.3 Input Encoding

It is not necessary to specify the ‘-C’ option if you use `\usepackage{isolatin1}` or `\documentstyle[isolatin1]{...}`.  $\text{\LaTeX2RTF}$  automatically detects these packages/style options and switches to processing of ISO-Latin1 codes. The following encodings are supported: ansinew, applemac, cp437, cp437de, cp850, cp852, cp865, decmulti, cp1250, cp1252, latin1, latin2, latin3, latin4, latin5, latin9, next, koi8-r, koi8-u, cp1251, cp855, cp866, maccyr, macukr, utf8, and utf8x. The encoding used in RTF files is cp1252. If cyrillic fonts are present, then these are represented in the RTF file using cp1251 (Windows Cyrillic).

### 4.4 Language Support

The following languages from the Babel package are supported: afrikaans, german, nynorsk, spanish, bahasa, dutch, icelandic, polish, swedish, basque, english, portuges, turkish, brazil, esperanto, irish, romanian, usorbian, breton, estonian, italian, samin, welsh, catalan, finnish, latin, scottish, croatian, lsorbian, serbian, czech, french, magyar, slovak, danish, galician, norsk, slovene.

The only thing that these files do is to translate various words usually emitted by  $\text{\LaTeX}$  during processing. For example, this ensures that the  $\text{\LaTeX2RTF}$  will provide the correct translation of the word “Chapter” in the converted document.

You can select any of the above languages using the ‘-l’ option. This is not needed if your  $\text{\LaTeX}$  file contains `\usepackage[language]{babel}`.

Encountering the ‘german’ package or documentstyle option (by H. Partl of the Viena University) makes  $\text{\LaTeX2RTF}$  behave like that: German Quotes, German Umlauts by “a, etc. . . This support is programmed directly into  $\text{\LaTeX2RTF}$  and supporting similar features for other languages will require patching the source code.

There is similar support for ‘french’ packages.

There is reasonable support for english, latin1, latin2, and cyrillic languages.

See [Section 5.7 \[Language Configuration\], page 23](#), for details on how to write a ‘language.cfg’ file for your language by yourself.

## 4.5 Cross References

Cross references include everything that you might expect and then some: bibliographic citations, equation references, table references, figure references, and section references. Section, equation, table and figure references are implemented by placing RTF bookmarks around the equation number (or table number or figure number).

Page references work but are implemented as “warm” cross-references. This means that Word does not automatically update the page references when the file is opened. To update the page references you must select the entire document (in Word) and press F9.

Bibliographic references currently require that a valid ‘.aux’ file be present. This is where LaTeX2RTF obtains the reference numbers. It would be nice if LaTeX2RTF just automatically numbered the references when there was no ‘.aux’ file, but LaTeX2RTF does not do this yet.

LaTeX2RTF relies on BibTeX to convert and format bibliographic entries. Usually the style file for a particular BibTeX format does not use any special LaTeX commands and therefore the bibliography file ‘file.bbl’ can be processed by LaTeX2RTF without difficulty. As a consequence, LaTeX2RTF can handle most bibliography styles without problem.

There are several latex style packages that add additional latex commands to enhance bibliographic formatting. LaTeX2RTF currently supports the following bibliographic packages:

apacite, apalike, authordate, harvard, natbib (also with apanat1b). These packages have many, many options and you may encounter problems with formatting in special cases.

As of LaTeX2RTF 1.9.17, the natbib command bibpunct is supported.

Footnotes are implemented and appear at the bottom of each page.

Indexing is reasonably-well supported. The simple mark-up of `makeindex`

```
\index{topic!subtopic@textit{subtopic}}
```

is supported. The rest of the fancy indexing stuff is not implemented. The index is created at the location of the `\printindex` command. When a file with an index is first opened in Word, you must select the entire file and update the page references and fields by pressing F9.

Currently, there is no support for `\labels` of `\items` in enumerate environments.

The conversion of cross-references is not perfect because of the different mechanisms in the LaTeX and Word worlds. In particular, if there are multiple `\label` in a figure, table, or section environment then only the first gets processed. It is also possible to confuse the LaTeX2RTF in eqnarray environments.

## 4.6 Page Formatting

$\text{\LaTeX2RTF}$  will handle some basic page formatting options, including `\doublespacing` (as implemented in the `setspace` package), and the margin setting options provided by the `geometry` package including commands in the `ratio`, `centering` and `margin` families. Not all `geometry` options are implemented yet, in part because there are no corresponding `rtf` commands for many of them.

## 4.7 Equations

There are five separate levels of equation translation based on the `-M` switch, see [Section 3.2 \[LaTeX2RTF Options\]](#), page 9. Each equation is now converted either to an EQ field or to a bitmap or inserted as raw TeX code in the document.

This is an interim solution (for some definition of “interim”). Ideally the equations would become OLE equation objects in the RTF file, but this needs to be implemented.

Some functions in the EQ fields have two or more parameters with a separator between each two. Unfortunately, the interpretation of these separators depends on the country specific settings in the MS Windows system in which the `rtf` file is opened. E.g. in English versions of MS Windows, the default parameter separator is the comma, in German versions the default is the semicolon. If the parameter in the RTF file does not match the Windows setting, some EQ fields are not interpreted correctly. You can check and set the separator in [Windows control panel - country settings - numbers - list separator]. By default,  $\text{\LaTeX2RTF}$  uses the comma as separator. If  $\text{\LaTeX2RTF}$  is called with the command line parameter `-S`, the semicolon is inserted as parameter delimiter.

Theoretically, according to the Word help file, parentheses ‘(’ or ‘)’ in mathematical formulas should be escaped (by a preceding backslash). Despite this, adding escapes usually produces a worse result than doing nothing. If Word displays some formulas with parentheses as “Error!”, you might try the `-p` option as a last resort.

It is also possible to convert an EQ field generated by  $\text{\LaTeX2RTF}$  to an Equation Editor object by opening the `rtf` file in Word and double-clicking on the equation. However there are bugs in the interface between Word and Equation Editor which prevent symbols in font Symbol or MTExtra to be converted correctly. The full commercial version of the Equation Editor, called MathType, handles this conversion correctly. If you have MathType version 6, an even better way to convert  $\text{\LaTeX}$  equations to MathType is letting  $\text{\LaTeX2RTF}$  write the  $\text{\LaTeX}$  code of the equations verbatim in the `rtf` file (option `-M32`), then open the `rtf` file in Word, select the  $\text{\LaTeX}$  code of an equation, cut it to the clipboard, open MathType, and paste the code. MathType will convert the code into an equation. Wilfried tried to automate this in a Word macro, but this fails because the macro does not wait until MathType is started and ready to receive the pasted code. Maybe the MathType authors will give us a hint or provide such a macro.

MathType can be downloaded from <http://www.dessci.com/en/> (30 day test version).

## 4.8 Math and Special Symbols

The way that symbols are converted in the RTF is based on the following observations. If the symbol is found in the latin 1 character set, then the current font is used. The font “Symbol” is widely available, and therefore it is used for as many translations as possible.

The font “MT Extra” is less common, but is free (as in beer from Design Science, see below) and characters from this font are used when possible. Note that early versions of “MT Extra”, including the one which is installed if you choose to install the Equation Editor coming with MS Word and MS Office, are missing some characters; the most recent version of this font (from Design Science) should be installed for best results. Finally, if the symbol is not found in any of the previous fonts, then `LATEX2RTF` will use the unicode sequence for that symbol.

This means that on the system where the `.rtf` file is opened, a unicode font should be available or these glyphs will not be displayed correctly. On Windows machines the unicode font is “Lucida Sans Unicode”, on MacOS X the font is “Lucida Grande”. Even when these fonts are installed, support for these unicode sequences seems to be hit-or-miss in various versions of Word since many of these symbols are not (yet) contained in the standard unicode fonts.

Many mathematical and special symbols are directly supported by `LATEX2RTF`. Less common symbols (not found in the fonts “Symbol” or “MT Extra”) are supported by the conversion table in the file `direct.cfg`, see [Section 5.4 \[Direct Conversion\]](#), page 22. An alternative `direct.cfg` is available in which the symbols are explicitly taken from “Lucida Sans Unicode”, which contains more of the symbols than the standard font Times / Times New Roman but which is only available on PCs with MS Windows. To use “Lucida Sans Unicode”, rename `direct.cfg` (to e.g., `direct_mt.cfg`) and rename `direct_ucs.cfg` to `direct.cfg`.

Required fonts are:

- “Times” / “Times New Roman”, preferably with Unicode extension (i.e. supporting all European languages including Cyrillic, Greek, and Hebrew). “Times” or “Times New Roman” is standard on all systems, but not its Unicode extension.
- “Symbol”, which is standard on all systems.
- “MT Extra”. This font is installed with the Microsoft Equation Editor, which comes with Microsoft Word, or its full version MathType. If you don’t have Word or MathType, you can get this font from <http://www.mathtype.com/en/dl/fonts/>

## 4.9 Tables

Conversion of tabular and tabbing environments is somewhat lame. The main difficulty is that `LATEX` (and `html`) will size the columns of a table automatically. There is no such feature in RTF. Consequently, the conversion defaults to making all the columns have equal size. This is suboptimal and should be revised.

Another way is to use the option `-t2` to make latex render them as bitmaps and insert the bitmaps into the RTF file. This feature was added in version 1.9.19.

## 4.10 Graphics

There is now rudimentary support for `\includegraphics`. Three file types will be inserted into the RTF file without needing conversion: `.pict`, `.jpeg`, and `.png` files. EPS and PDF files are converted to PNG using `convert` from the ImageMagick package. Some options are even handled properly.

## 4.11 Pagestyles

If there is no `\pagestyle` command, the RTF output is generated as with plain pagestyle, i.e. each page has its page number centered at the bottom.

You must turn this off with the `\pagestyle{empty}` command in the  $\LaTeX$  file if you don't want pagenumbers. The headings and myheadings styles are silently ignored by now. The twosided option to the `\documentstyle` or `\documentclass` produces the corresponding RTF tokens. Note that these features require RTF Version 1.4.

## 4.12 Hyperlatex

Hyperlatex support is largely broken at the moment, but continues to improve.

Otfried Schwarzkopf has created the “Hyperlatex Markup Language” which is a “little package that allows you to use  $\LaTeX$  to prepare documents in HTML.” It brings an Emacs lisp program with it to convert the Hyperlatex file to HTML. Hyperlatex can be obtained from the CTAN-sites, see [Section 2.2 \[Obtaining LaTeX2RTF\], page 3](#). There are two convenient commands that avoid typing: `\link` and `\xlink` that generate an “internal” label which then is used in the following `\Ref` and `\Pageref` commands.

$\LaTeX$  makes it possible to write `'\link{anchor}[ltx]{label}'`, which typesets: ‘anchor ltx’.  $\LaTeX2RTF$  does NOT support this aproach since the optional parameter is thrown away right now, see [Chapter 8 \[LaTeX2RTF under Development\], page 29](#).

Note that you have to update your ‘.cfg’ files if you are upgrading, since there are a lot of HTML oriented commands in Hyperlatex that we simply can ‘ignore’.



## 5 Configuration

### 5.1 Input processing

On processing input `LATEX2RTF` first converts the `LATEX` special characters. If it encounters one of the standard commands it is converted internally. If a command is not known to `LATEX2RTF` it is first looked up in ‘`direct.cfg`’ and the RTF code specified there is output. If not found there it is looked up in the section ‘`ignore.cfg`’. This file includes a lot of `LATEX` commands that do not affect the output (cross reference information and the like), or that we are not able or willing to convert to RTF.

You can use ‘`ignore.cfg`’ if you get tired of seeing

```
WARNING: command: ‘foo’ not found - ignored
```

and you don’t need ‘foo’ in your RTF document. It would be nice to send your additions to the `LATEX2RTF` mailing list for inclusion in later distributions.

`LATEX2RTF` accepts Unix, MS-DOS, and Macintosh line ending codes (`\n`, `\r\n` and `\r`). The files it creates get the line ending for the platform on which `LATEX2RTF` was compiled.

The `LATEX` file may have been created with a wide variety of character sets. If the `LATEX` lacks the `\package[codepage]{inputenc}` definition, then you may need to use the command line switch to manually select the proper code page (see [Section 4.3 \[Input Encoding\]](#), page 15).

### 5.2 Conditional Parsing

Starting with `LATEX2RTF` 1.9.18, there is a handy method for controlling which content should be processed by `LATEX` or by `LATEX2RTF`. Control is achieved using the standard `\if` facility of `TEX`. If you include the following line in the preamble of your document (i.e., before `\begin{document}`)

```
\newif\iflatextortf
```

Then you will create a new `\iflatextortf` command in `LATEX`. `TEX` sets the value of this to `false` by default. Now, `LATEX2RTF` internally sets `\iflatextortf` to be true, and to ensure that this is always the case, `LATEX2RTF` ignores the command `\latextortffalse`. This means that you can control how different applications process your document by

```
\iflatextortf
This code is processed only by latex2rtf
\else
This code is processed only by latex
\fi
```

Note that `\iflatextortf` will only work within a section; you cannot use this command to conditionally parse code that crosses section boundaries. Also, it will only work on complete table or figure environments. Due to the mechanism used by `LATEX2RTF` in processing these environments, at this time the only way to conditionally parse tables and figures is to include two complete versions of the environment in question, nested within an appropriate `\iflatex2rtf` structure.

LaTeX2RTF versions 1.9.15 to 1.9.18 had the ability to hide contents from LaTeX but expose them to LaTeX2RTF by starting a line with `%latex2rtf:`. This code was horribly broken, and it was removed. The same functionality is readily achieved using the `\iflargetortf` mechanism. For example, the old method allowed

```
%latex2rtf: This line will only appear in the latex2rtf output,
```

To get the same behavior, define `\iflargetortf` and use

```
\iflargetortf
This code is processed only by latex2rtf
\fi
```

### 5.3 Output Formatting

On writing output, LaTeX2RTF generates the operating system specific line ending code (`\n` on Unix, `\r\n` on DOS or Windows), depending on which system LaTeX2RTF was compiled. As both should be legal to any RTF Reader the resulting RTF rendering should not be affected.

LaTeX2RTF does not offer a whole lot of flexibility in how files are translated, but it does offer some. This flexibility resides in four files `'direct.cfg'`, `'ignore.cfg'`, `'fonts.cfg'`, and `'language.cfg'`. These files are documented in the next four sections.

### 5.4 Direct Conversion

The file `'direct.cfg'` is used for converting LaTeX commands by simple text replacement. The format consists of lines with a LaTeX command with backslash followed by comma. The rest of the line until a `'.'` character will be written to the RTF file when the command is found in the LaTeX file. Lines starting with a `'#'` character are ignored. After the `'.'` everything is ignored to end of line. To select a specific font use `*fontname*`, where `fontname` be defined in `'fonts.cfg'`. To write the `'*'` character use `**`.

```
\bigstar,{\u8727**}.
\copyright,\'a9.
```

In general, specific fonts should not be specified in this file. There is a mechanism to do this, but it turns out that this is not as useful as originally thought. The main reason that this fails is because the conversion of equations from Word fields to Equation Editor objects is buggy. The consequence is that to have symbols show up properly, they must be encoded differently when the Symbol and MT Extra fonts are used — depending on whether Word fields are active or not. It was all very tedious to figure out a mechanism that was “least broken.”

### 5.5 Ignore Command

The file `'ignore.cfg'` is used for defining how to ignore specific commands. This file is used for recognition of LaTeX variables, user defined variables, and some simple commands. All variables are ignored but the converter must know the names to correctly ignore assignments to variables. Lines in this file consist of a variable name with backslash, followed by comma and the type of the variable followed by `'.'`. Possible types are

```
'NUMBER'    simple numeric value
```



‘MEASURE’	numeric value with following unit of measure
‘OTHER’	ignores anything to the first character after ‘=’ and from there to next space. e.g., <code>\setbox\bak=\hbox</code>
‘COMMAND’	ignores anything to next ‘\’ and from there to the occurrence of anything but a letter e.g., <code>\newbox\bak</code>
‘SINGLE’	ignore single command e.g., <code>\noindent</code>
‘PARAMETER’	ignores a command with one parameter e.g., <code>\foo{bar}</code>
‘PACKAGE’	does not produce a Warning message if PACKAGE is encountered, e.g., ‘PACKAGE,kleenex.’
‘ENVCMD’	processes contents of unknown environment as if it were plain $\text{\LaTeX}$ eg. ‘ENV-CMD,envir.’ Therefore <code>\begin{envir} text \end{envir}</code> as ‘text’.
‘ENVIRONMENT’	ignores contents of that environment, e.g., with ‘ENVIRONMENT,ifhtml.’ <code>\begin{ifhtml} text \end{ifhtml}</code> ignores ‘text’.

The types are in upper case exactly as above. Do not use spaces. Lines starting with a ‘#’ character are ignored. After the ‘.’ everything is ignored to end of line. Example:

```
\pagelength,MEASURE.
```

## 5.6 Font Configuration

The file ‘`fonts.cfg`’ contains the font name mapping. For example, this file determines what font is used to represent `\rm` characters in the RTF file.

A line consists of a font name in  $\text{\LaTeX}$  followed by comma and a font name in RTF. The end is marked by a ‘.’. No spaces are allowed. The  $\text{\LaTeX}$  font will be converted to the RTF font when it is found in the  $\text{\LaTeX}$  file. If multiple translations for the same  $\text{\LaTeX}$  font are specified, only the first is used. All fonts in a  $\text{\LaTeX}$  file that are not in this file will be mapped to the default font. All RTF fonts listed in this file will be in every RTF file header whether used or not. Lines starting with a ‘#’ character are ignored. After the ‘.’ everything is ignored to end of line.

To add a RTF font not used as substitute for a  $\text{\LaTeX}$  font — for example a Symbol font used in ‘`direct.cfg`’ — use a dummy  $\text{\LaTeX}$  name like in the following

```
Dummy3,MathematicalSymbols.
```

Make sure you use the correct font name. Take care of spaces in font names. The default fonts are named Roman `\rm`, Slanted `\sl`, Sans Serif `\sf`, Typewriter `\tt`, or Calligraphic `\cal`.

## 5.7 Language Configuration

The file(s) ‘`language.cfg`’ control the translation of  $\text{\LaTeX}$ ’s “hardcoded” sectioning names. The standard  $\text{\LaTeX}$  styles have some fixed Title names like ‘Part’, ‘Reference’ or ‘Bibliography’ that appeared in English or German in the output with the original versions of  $\text{\LaTeX}2\text{RTF}$ .

It is unlikely that you will need to create a new ‘`language.cfg`’ file. However, just look at one of the existing files and follow the pattern. The format is really simple.

## 6 Error Messages and Logging

As stated in the Debugging section, `LATEX2RTF` provides a means to control the amount of debugging information through the ‘`-d#`’ switch. By using a debugging level of 4, you can get a pretty good idea of what `LATEX` command caused the problem and what line that command might be found on.

### ‘Fatal error messages’

indicate a bug in the source code. PLEASE report them, if they do not appear in the documentation (see [Section 8.4 \[Reporting Bugs\]](#), page 29).

### ‘Error messages’

always abort the program and are caused by conditions that prevent further conversion of the input file. Typically this is caused by `LATEX2RTF` getting hopelessly confused by the number of braces in the `LATEX` file.

### ‘Warning messages’

inform you, that there is some conversion loss from `LATEX` to RTF, or that the output file has some restrictions on some RTF Readers. Most of these warnings can be suppressed by adding the offending command to the ‘`ignore.cfg`’ file.

Error and Warning messages should follow the GNU Coding standards, i.e. they have the format

```
inputfile':line: Error|Warning: message
```

You can also control the level of debugging output by inserting `\verbositylevel{#}` in the `LATEX` file. This is very handy if you have a large `LATEX` file that is failing in only a small section. For example,

```
problem free latex file ....
\verbositylevel{5}
problematic code
\verbositylevel{0}
```

will cause a huge amount of debugging information to be emitted for the problematic code.

Error reporting and logging still has many inconsistencies, but it gets better with each release. Don't try to make any sense in debugging levels above 4, these are for my own delight only and can change significantly between versions.

The ‘`inputfile`’ may be incorrectly identified if it is incorporated through `\input` or `\include`. The line may be also be wrong at times. See [Section 8.3 \[Known Bugs\]](#), page 29.



## 7 History & Copyright

In 1994 the first Version of  $\text{\LaTeX}2\text{RTF}$  was written by Fernando Dorner and Andreas Granzer of the Viena University supervised by Ralf Schlatterbeck in a one-semester course. They created a simple  $\text{\LaTeX}$  parser and added most of the infrastructure for the program. This was version 1.0 of  $\text{\LaTeX}2\text{RTF}$ . In 1995, work on  $\text{\LaTeX}2\text{RTF}$  was continued in another one-semester course by Friedrich Polzer and Gerhard Trisko. The result was  $\text{\LaTeX}2\text{RTF}$  version 1.5. Ralf Schlatterbeck (ralf “at” zoo.priv.at) maintained and extended  $\text{\LaTeX}2\text{RTF}$  until 1998.

In 1998 Georg Lehner (jorge.lehner “at” gmx.net) found the reference to  $\text{\LaTeX}2\text{RTF}$  on the [TeX Conversion Webpage](#) of Wilfried Hennings and added some functionality and took over the maintainence of the program. The last version release by Georg is 1.8aa. The bulk of development post 1.8aa was done by Scott Prahl. Wilfried Hennings now coordinates the development of the program and maintains the project on [SourceForge](#) where there are also (low volume) mailing lists for users and developers. Mailing via one of these lists requires subscription to the list (to prevent spam). For subscription to these lists visit the page: [users list](#) or [developers list](#)

As of November 2007, version 1.9.19 of  $\text{\LaTeX}2\text{RTF}$  is available. One day there shall be a jump to Version 2.0, but this is not history but future ...

The contents of this manual were composed by copying shamelessly what was available in the original sources and documentation.



## 8 LaTeX2RTF under Development

### 8.1 Unimplemented Features

- LaTeX2RTF ignores some optional parameters of `\documentstyle`
- Add the code to produce the corresponding chapter, section, and page numbering with headings and myheadings pagestyles. Implement `\markboth` and `\markright`.
- To support `\tableofcontents` there would be two approaches: Transfer sectioning information, title text and then produce page numbers by the rtf- reader. Scan and label all of the sectioning commands while reading and then construct the sectioning information using these labels. Needs two passes on LaTeX input.

### 8.2 Missing options

Missing or buggy command line options.

‘-d’            Information logging and Error reporting is not implemented consistently. Need to test and track problems with the linenumber and with the file name.

‘--long\_names’            It would be useful to implement the GNU long option names, e.g.: ‘-debug’, ‘-output\_file’, ‘-quiet’, etc. This could be done by switching to the GNU getopt package.

### 8.3 Known Bugs

1. The first parameter of a `\link{anchor}[ltx]{label}` is converted to the rtf-output. Label is stored to `hyperref` for later use, the optional parameter is ignored. [ltx] should be processed as Otfried recommends it, to use for exclusive LaTeX output.e.g: `\link{readhere}[~\Ref]{explaining: chapter}`. Since `{explaining:chapter}` is yet read by LaTeX and hyperlatex when [...] is evaluated it produces the correct reference. LaTeX2RTF is only strolling from left to right through the text and can’t remember what she will see in the future.
2. The diagnostics routine does not output the correct (actual) input filename. (‘.aux’, ‘.bbl’, `\input`).

### 8.4 Reporting Bugs

Report bugs to to the bug tracking system at [SourceForge](https://sourceforge.net/projects/latex2rtf/). Only report bugs for the latest version of LaTeX2RTF that is available. Please provide the following information and observe the following guidelines when reporting a bug in the program:

1. State the version of LaTeX2RTF that you are using. You can get the version by specifying the ‘-V’ option to LaTeX2RTF .
2. Specify the your operating system and version. Be sure to check the file ‘Makefile’ for settings that may be specific to your machine, especially for some versions of SunOS there may be settings which are needed to compile successfully. Do this before submitting a bug report.

3. If the program produces wrong output or does not work for you, include a short  $\text{\LaTeX}$  file along with a description of the problem. Isolating the bug into a small  $\text{\LaTeX}$  file does two things. First, it provides a file that can be used to test future versions of  $\text{\LaTeX2RTF}$  and second, it certainly improves the chances that the bug will get some attention. Do not send me large  $\text{\LaTeX}$  or RTF files, I simply do not have the time to wade through large files to search for a bug!
4. Be patient. I am maintaining the program in my free time. I did not write most of the code. Often I do not have the time to answer to your question. I will, however, try to fix reported bugs in upcoming releases.

## 8.5 Todo List

Scott's ToDo list

- Use lex/yacc to implement getSection
- Add support for pagestyle
- Better support for ignoring commands

Georg's todo list

- Make this Manual more consistent, the ToDo and Known Bug List shorter and the Features List longer.
- Harmonize all of the error and warning messages.
- Put warnings everywhere applicable about producing RTF 1.4 tokens.
- Provide an Error and Warning recovery guide to the user.
- Add a chapter with lists of all  $\text{\LaTeX}$  commands that convert, and that do not convert to RTF, including their status (for future releases, never, partially functional, ...).

## 8.6 Command List

Listed here are all the  $\text{\LaTeX}$  commands currently parsed by  $\text{\LaTeX2RTF}$ . Note: inclusion in this list does not mean that a command is fully and correctly handled by  $\text{\LaTeX2RTF}$ . In some cases the commands here are place-holders only, and are not implemented at all. The list is provided to encourage developers to note any departures from the behaviour that  $\text{\LaTeX}$  users will expect. The location of the commands is noted to assist anyone interested in hacking on the C source code. This list is a work in progress, and may not be immediately useful to general users, other than to indicate those commands that we have at least contemplated implementing.

### 8.6.1 General Commands

These commands are found in the `commands[]` array in `commands.c`. They are arranged alphabetically within sections according to function.

#### 8.6.1.1 Basic Commands

All listed commands work as expected.

`begin`

`centerline`



end  
endnote  
footnote  
raggedright  
the  
today  
vcenter

### 8.6.1.2 Font Commands

All listed commands work as expected.

bf  
bfseries  
cal  
em  
emph  
enotesize  
footnotesize  
HUGE  
Huge  
huge  
it  
itshape  
LARGE  
Large  
large  
mathbf  
mathcal  
mathit  
mathmd  
mathnormal  
mathrm  
mathsc  
mathsf  
mathsl

mathtt  
mathup  
mdseries  
mit  
normalfont  
normalsize  
rm  
rmfamily  
sc  
scfamily  
scriptsize  
scshape  
sf  
sffamily  
sl  
slshape  
small  
ssmall  
textbf  
textfont  
textit  
textmd  
textnormal  
textrm  
textsc  
textsf  
textsl  
texttt  
textup  
tiny  
tt  
ttfamily  
underbar  
underline  
upshape

### 8.6.1.3 Logos

All listed commands work as expected.

AmSLaTeX

AmSTeX

BibTeX

kern

LaTeX

latex

LaTeXe

lower

LyX

SLiTeX

TeX

### 8.6.1.4 Special Characters

These commands all work as expected.

\

acute

b

bar

breve

c

check

d

ddot

dot

grave

H

hat

i

j

l

L

r

tilde

u

v

vec

### 8.6.1.5 Sectioning Commands

chapter

chapter\*

paragraph

paragraph\*

part

part\*

section

section\*

subparagraph

subparagraph\*

subsection

subsection\*

subsubsection

subsubsection\*

### 8.6.1.6 Uncategorized

These commands need to be organized into new or existing sections.

abstract

addcontents

Ignored

addcontentsline

Ignored

addvspace Ignored

aleph

Alph

alph

Ignored

alpha

Alpha

Ignored

amalg

and

angle

appendix Ignored

approx

arabic Ignored

ast

author

baselineskip

because

beta

Beta

bibentry

bibitem

bibliography

bibliographystyle

bibliographystyle  
Ignored

bigskip

bot

BoxedEPSF

bullet

cap

caption

cdot

cdots

centering

char

chi

Chi

circ

cite

citeonline

cleardoublepage

clearpage

clubsuit

cong

contentsline

coprod

cup

date

ddots

delta

Delta

dfrac

Diamond

diamondsuit

div

doteq

dotfill      There is no rtf code for dotfill;  $\text{\LaTeX2RTF}$  inserts an ellipsis only.

dots

dots

downarrow

Downarrow

efloatseparator

ell

emptyset

endinput

endnotemark

Ignored

ensuremath

epsfbox

epsfile

epsilon

eqref

equiv

eta

exists

fbox

fbox

fnsymbol    Ignored

footnotemark

Ignored

forall

frac

Frac

framebox Ignored

frenchspacing

Ignored

gamma

Gamma

ge

geq

gg

glossary Ignored

glossaryentry

Ignored

hbar

hbox

heartsuit

hsize

hslash

hspace Ignored

hspace\* Ignored

htmladdnormallink

htmlref

iiint

iint

Im

in

include

includegraphics

includegraphics\*

includeonly

Ignored

indent

index

indexentry  
Ignored

infty

input

int

int

iota

kappa

label

lambda

Lambda

land

langle

lceil

ldots

le

left

leftarrow

Leftarrow

leftharpoondown

leftleftarrows

leftrightarrow

Leftrightarrow

leftrightharpoons

leq

let Ignored

letterspace

lfloor

lim

liminf

limsup

linebreak Ignored

lineskip

listoffigures

listoftables

ll



longleftarrow  
longleftarrows  
longrightarrow  
lor  
makebox Ignored  
maketitle  
mapsto  
marginpar Ignored  
markboth Ignored  
markright Ignored  
matrix  
mbox  
measuredangle  
medskip  
mho  
moveleft  
moveright  
mp  
mu  
multicolumn  
nabla  
ne  
narrow  
neg  
neq  
newblock  
newcount Ignored  
newfont Ignored  
newpage  
newsavebox  
Ignored  
nobibliography  
Ignored  
nobreakspace  
nocite

noindent

nolinebreak

Ignored

nonfrenchspacing

Ignored

nonumber

nopagebreak

Ignored

notag

nu

numberline

narrow

omega

Omega

omicron

onecolumn

onlinecite

oplus

oslash

otimes

output

Ignored

overline

pagebreak

pagenumbering

Ignored

pageref

pagestyle

Ignored

par

parbox

partial

perp

phi

Phi

pi

Pi

pm  
prec  
printindex  
prod  
prod  
propto  
protect Ignored  
psfig  
psfrag Ignored  
psi  
Psi  
qquad  
quad  
raisebox Ignored  
rangle  
rceil  
Re  
ref  
refstepcounter  
rfloor  
rho  
right  
Rightarrow  
rightarrow  
rightharpoonup  
rightleftarrows  
rightleftharpoons  
rightrightarrows  
Roman Ignored  
roman Ignored  
rule  
samepage Ignored  
savebox Ignored  
sbox Ignored  
searrow

setbox  
settowidth Ignored  
sigma  
Sigma  
sim  
simeq  
smallskip  
spadesuit  
sqrt  
stackrel  
stepcounter  
Ignored  
stretch Ignored  
subset  
subseteq  
succ  
sum  
sum  
supset  
supseteq  
surd  
swarrow  
tableofcontents  
tau  
textalpha  
textbeta  
textbullet  
textchi  
textcolor  
textDelta  
textdelta  
textellipsis  
textepsilon  
texteta

textGamma  
textgamma  
textiota  
textkappa  
textLambda  
textlambda  
textmu  
textnu  
textOmega  
textomega  
textperiodcentered  
textPhi  
textphi  
textPi  
textpi  
textPsi  
textpsi  
textSigma  
textsigma  
textsubscript  
textsuperscript  
textTau  
texttau  
textTheta  
texttheta  
textXi  
textxi  
textzeta  
thanks  
therefore  
Theta  
theta  
times  
title  
to

triangleleft  
triangleright  
twocolumn  
typeout Ignored  
Typein Ignored  
typein Ignored  
typeout Ignored  
Uparrow  
uparrow  
updownarrow  
Updownarrow  
upsilon  
Upsilon  
url  
usebox Ignored  
value  
varepsilon  
varnothing  
varphi  
varpi  
varpropto  
varsigma  
vartheta  
vbox  
vdots  
vee  
verb  
verb\*  
vref  
vsize  
vskip  
vspace  
vspace\*  
wedge  
wp

xi

Xi

zeta

### 8.6.2 Preamble Commands

These commands are found in `PreambleCommands[]` in `commands.c`, and are implemented in `preamble.c`.

addtocounter

addtolength

baselineskip

celsius

cfoot

chead

cline

DeclareRobustCommand

DeclareRobustCommand\*

def

degreecelsius

documentclass

documentstyle

doublespacing

Currently, the only command from the `setspace` package that is implemented, and the only way to modify line spacing.

endnotetext

EUR

euro

evensidemargin

fancyfoot

fancyhead

flushbottom

footnotetext

`geometry` Currently recognizes `*ratio`, `*centering`, `*margin`, `left`, `right`, `inner`, `outer`, `top`, `bottom`, `right`, `left` (including `vmargin`, `hratio` etc.)

headheight

headsep

hline

hoffset

htmladdnormallink

htmlref

hyphenation  
iflaxtextortf  
ifx  
include  
input  
laxtextortffalse  
Ignored  
laxtextortftrue  
Ignored  
lfoot  
lhead  
listoffiles Ignored  
makeglossary  
Ignored  
makeindex  
Ignored  
makelabels  
Ignored  
markboth Ignored  
markright Ignored  
newcommand  
newcounter  
newenvironment  
newif  
newlength  
newtheorem  
nobreakspace  
nofiles Ignored  
oddsidemargin  
pagenumbering  
Ignored  
pagestyle  
parindent  
parskip  
providecommand  
raggedbottom  
renewcommand  
renewenvironment  
renewtheorem  
Ignored



resizebox  
resizebox\*  
rfoot  
rhead  
setcounter  
setlength  
signature  
textheight  
textwidth  
theendnotes  
thepage  
thispagestyle  
          Ignored  
topmargin  
usepackage  
verbositylevel  
voffset

### 8.6.3 Letter Commands

Found in `LetterCommands []` in `commands.c`.

address  
cc  
closing  
encl  
opening  
ps  
signature

### 8.6.4 Language Commands

#### 8.6.4.1 German Commands

Found in `GermanModeCommands []` in `commands.c`.

ck  
glqq  
glq  
grq  
grqq

### 8.6.4.2 Czech Commands

uv

### 8.6.4.3 French Commands

deuxpoints

dittomark

FCS

fg

fup

ieme

iemes

ier

iere

ieres

iers

inferieura

LCS

lq

lqq

numero

Numero

numeros

Numeros

og

pointexclamation

pointinterrogation

pointvirgule

primo

quarto

rq

rqq

secundo

superieura

tertio

up

#### 8.6.4.4 Russian Commands

CYRA

cyr a

CYRB

cyr b

CYRC

cyr c

CYRCH

cyr ch

CYRCHSH

cyr chsh

CYRD

cyr d

CYRE

cyr e

CYREREV

cyr erev

CYRERY

cyr ery

CYRF

cyr f

CYRG

cyr g

CYRH

cyr h

CYRHRDSN

cyr hrdsn

CYRI

cyr i

CYRISHRT

cyr ishrt

CYRK

cyr k

CYRL

cyr1

CYRM

cyrm

CYRN

cyrn

CYRO

cyro

CYRP

cyrp

CYRR

cyr r

CYRS

cyr s

CYRSFTSN

cyr sftsn

CYRSH

cyrsh

CYRT

cyr t

CYRU

cyr u

CYRV

cyr v

CYRYA

cyr ya

CYRYU

cyr yu

CYRZ

cyr z

CYRZH

cyrzh

### 8.6.5 Citation Commands

### 8.6.5.1 Apacite Commands

AX

BAnd

BBA

BBAA

BBAB

BBAY

BBC

BBCP

BBCQ

BBN

BBOP

BBOQ

BCAY

BCBL

BCBT

BCHAIR

BCHAIRS

BCnt

BCntIP

BED

BE<sub>d</sub>

BEDS

Bem

BIP

BMTh

BNUM

BNUMS

BOthers

BOWP

BPG

BPGS

BPhD

BREPR

BTR

BTRANS

BTRANSS

BUMTh

BUPhD

BVOL

BVOLS

citeA

citeauthor

citeNP

citeyear

citeyearNP

fullcite

fullciteA

fullciteauthor

fullciteNP

shortcite

shortciteA

shortciteauthor

shortciteNP

### **8.6.5.2 AuthorDate Commands**

citename

shortcite

### **8.6.5.3 Harvard Commands**

cite

citeaffixed

citeasnoun

citename

citeyear

citeyear\*

harvardand

harvarditem

harvardyearleft

harvardyearright

possessivecite

#### 8.6.5.4 HyperLatex Commands

Cite

link

Pageref

Ref

S

xlink

#### 8.6.5.5 Apacite Commands

bibpunct

cite

citealp

Citealp

citealp\*

citealt

Citealt

citealt\*

citeauthor

Citeauthor

citeauthor\*

citep

Citep

citep\*

citet

Citet

citet\*

citetext

citeyear

citeyearpar

#### 8.6.6 Other Commands

Other Commands:

item

caption

center

### 8.6.7 Environments

Environments processed - found in `params[]` in `commands.c`.

`abstract`

`acknowledgments`

`align`

`align*`

`alltt`

`array`

`bf`

`bfseries`

`center`

`comment`

`compactenum`

`compactitem`

`description`

`displaymath`

`document`

`em`

`enumerate`

`eqnarray`

`eqnarray*`

`equation`

`equation*`

`figure`

`figure*`

`flushleft`

`flushright`

`htmlonly` Ignored

`it`

`itemize`

`itshape`

`landscape`

`latexonly` Ignored

`letter`



list  
longtable  
longtable\*  
math  
mdseries  
minipage  
multicolumn  
music  
picture  
quotation  
quote  
rawhtml    Ignored  
rm  
rmfamily  
sc  
scshape  
sf  
sffamily  
sl  
sloppypar  
slshape  
small  
tabbing  
table  
table\*  
tabular  
tabular\*  
thebibliography  
theindex    Ignored  
titlepage  
tt  
ttfamily  
verbatim  
Verbatim  
verse



## 9 Function Index

<code>\</code>		
<code>\</code> .....	33	
<b>A</b>		
<code>abstract</code> .....	34, 54	
<code>acknowledgments</code> .....	54	
<code>acute</code> .....	33	
<code>addcontents</code> .....	34	
<code>addcontentsline</code> .....	34	
<code>address</code> .....	47	
<code>addtocounter</code> .....	45	
<code>addtolength</code> .....	45	
<code>addvspace</code> .....	34	
<code>aleph</code> .....	34	
<code>align</code> .....	54	
<code>align*</code> .....	54	
<code>alltt</code> .....	54	
<code>alph</code> .....	34	
<code>Alph</code> .....	34	
<code>alpha</code> .....	34	
<code>Alpha</code> .....	34	
<code>amalg</code> .....	34	
<code>AmSLaTeX</code> .....	33	
<code>AmSTeX</code> .....	33	
<code>and</code> .....	34	
<code>angle</code> .....	34	
<code>appendix</code> .....	35	
<code>approx</code> .....	35	
<code>arabic</code> .....	35	
<code>array</code> .....	54	
<code>ast</code> .....	35	
<code>author</code> .....	35	
<code>AX</code> .....	51	
<b>B</b>		
<code>b</code> .....	33	
<code>BAnd</code> .....	51	
<code>bar</code> .....	33	
<code>baselineskip</code> .....	35, 45	
<code>BBA</code> .....	51	
<code>BBAA</code> .....	51	
<code>BBAB</code> .....	51	
<code>BBAY</code> .....	51	
<code>BBC</code> .....	51	
<code>BBCP</code> .....	51	
<code>BBCQ</code> .....	51	
<code>BBN</code> .....	51	
<code>BBOP</code> .....	51	
<code>BBOQ</code> .....	51	
<code>BCAY</code> .....	51	
<code>BCBL</code> .....	51	
<code>BCBT</code> .....	51	
<code>BCHAIR</code> .....	51	
<code>BCHAIRS</code> .....	51	
<code>BCnt</code> .....	51	
<code>BCntIP</code> .....	51	
<code>because</code> .....	35	
<code>BEd</code> .....	51	
<code>BED</code> .....	51	
<code>BEDS</code> .....	51	
<code>begin</code> .....	30	
<code>Bem</code> .....	51	
<code>beta</code> .....	35	
<code>Beta</code> .....	35	
<code>bf</code> .....	31, 54	
<code>bfseries</code> .....	31, 54	
<code>bibentry</code> .....	35	
<code>bibitem</code> .....	35	
<code>bibliography</code> .....	35	
<code>bibliographystyle</code> .....	35	
<code>bibpunct</code> .....	53	
<code>BibTeX</code> .....	33	
<code>bigskip</code> .....	35	
<code>BIP</code> .....	51	
<code>BMTh</code> .....	51	
<code>BNUM</code> .....	51	
<code>BNUMS</code> .....	51	
<code>bot</code> .....	35	
<code>BOthers</code> .....	51	
<code>BOWP</code> .....	51	
<code>BoxedEPSF</code> .....	35	
<code>BPG</code> .....	51	
<code>BPGS</code> .....	51	
<code>BPhD</code> .....	51	
<code>BREPR</code> .....	52	
<code>breve</code> .....	33	
<code>BTR</code> .....	52	
<code>BTRANS</code> .....	52	
<code>BTRANSS</code> .....	52	
<code>bullet</code> .....	35	
<code>BUMTh</code> .....	52	
<code>BUPhD</code> .....	52	
<code>BVOL</code> .....	52	
<code>BVOLS</code> .....	52	
<b>C</b>		
<code>c</code> .....	33	
<code>cal</code> .....	31	
<code>cap</code> .....	35	
<code>caption</code> .....	35	
<code>cc</code> .....	47	
<code>cdot</code> .....	35	
<code>cdots</code> .....	35	
<code>celsius</code> .....	45	
<code>center</code> .....	54	
<code>centering</code> .....	35	

centerline.....	30	CYRCH.....	49
cfoot.....	45	cyrchsh.....	49
chapter.....	34	CYRCHSH.....	49
chapter*.....	34	cyrd.....	49
char.....	35	CYRD.....	49
chead.....	45	cyre.....	49
check.....	33	CYRE.....	49
chi.....	35	cyrerev.....	49
Chi.....	35	CYREREV.....	49
circ.....	35	cyrery.....	49
cite..... 35, 52,	53	CYRERY.....	49
Cite.....	53	cyrf.....	49
citeA.....	52	CYRF.....	49
citeaffixed.....	52	cyrg.....	49
citealp.....	53	CYRG.....	49
Citealp.....	53	cyrh.....	49
citealp*.....	53	CYRH.....	49
citealt.....	53	cyrhrdsn.....	49
Citealt.....	53	CYRHRDSN.....	49
citealt*.....	53	cyri.....	49
citeasnoun.....	52	CYRI.....	49
citeauthor..... 52,	53	cyrishrt.....	49
Citeauthor.....	53	CYRISHRT.....	49
citeauthor*.....	53	cyrk.....	49
citename.....	52	CYRK.....	49
citeNP.....	52	cyrl.....	50
citeonline.....	35	CYRL.....	49
citep.....	53	cyrn.....	50
Citep.....	53	CYRN.....	50
citep*.....	53	cyro.....	50
citet.....	53	CYRO.....	50
Citet.....	53	cyrp.....	50
citet*.....	53	CYRP.....	50
citetext.....	53	cyrr.....	50
citeyear..... 52,	53	CYRR.....	50
citeyear*.....	52	cyrs.....	50
citeyearNP.....	52	CYRS.....	50
citeyearpar.....	53	cyrsftsn.....	50
ck.....	47	CYRSFTSN.....	50
cleardoublepage.....	35	cyrsh.....	50
clearpage.....	35	CYRSH.....	50
cline.....	45	cyrt.....	50
closing.....	47	CYRT.....	50
clubsuit.....	35	cyru.....	50
comment.....	54	CYRU.....	50
compactenum.....	54	cyrv.....	50
compactitem.....	54	CYRV.....	50
cong.....	35	cyrya.....	50
contentsline.....	36	CYRYA.....	50
coprod.....	36	cyryu.....	50
cup.....	36	CYRYU.....	50
cyra.....	49	cyrz.....	50
CYRA.....	49	CYRZ.....	50
cyrb.....	49	cyrzh.....	50
CYRB.....	49	CYRZH.....	50
cyrc.....	49		
CYRC.....	49		
cyrch.....	49		

**D**

d.....	33
date.....	36
ddot.....	33
ddots.....	36
DeclareRobustCommand.....	45
DeclareRobustCommand*.....	45
def.....	45
degrecelsius.....	45
delta.....	36
Delta.....	36
description.....	54
deuxpoints.....	48
dfrac.....	36
Diamond.....	36
diamondsuit.....	36
displaymath.....	54
dittomark.....	48
div.....	36
document.....	54
documentclass.....	45
documentstyle.....	45
dot.....	33
doteq.....	36
dotfill.....	36
dots.....	36
doublespacing.....	45
downarrow.....	36
Downarrow.....	36

**E**

efloatseparator.....	36
ell.....	36
em.....	31, 54
emph.....	31
emptyset.....	36
encl.....	47
end.....	31
endinput.....	36
endnote.....	31
endnotemark.....	36
endnotetext.....	45
enotesize.....	31
ensuremath.....	36
enumerate.....	54
epsfbox.....	36
epsffile.....	36
epsilon.....	36
eqnarray.....	54
eqnarray*.....	54
eqref.....	36
equation.....	54
equation*.....	54
equiv.....	36
eta.....	36
EUR.....	45
euro.....	45

evensidemargin.....	45
exists.....	36

**F**

fancyfoot.....	45
fancyhead.....	45
fbox.....	36
FCS.....	48
fg.....	48
figure.....	54
figure*.....	54
flushbottom.....	45
flushleft.....	54
flushright.....	54
fnsymbol.....	36
footnote.....	31
footnotemark.....	36
footnotesize.....	31
footnotetext.....	45
forall.....	37
frac.....	37
Frac.....	37
framebox.....	37
frenchspacing.....	37
fullcite.....	52
fullciteA.....	52
fullciteauthor.....	52
fullciteNP.....	52
fup.....	48

**G**

gamma.....	37
Gamma.....	37
ge.....	37
geometry.....	45
geq.....	37
gg.....	37
glossary.....	37
glossaryentry.....	37
glq.....	47
glqq.....	47
grave.....	33
grq.....	47
grqq.....	47

**H**

H.....	33
harvardand.....	52
harvarditem.....	52
harvardyearleft.....	52
harvardyearright.....	52
hat.....	33
hbar.....	37
hbox.....	37
headheight.....	45

headsep	45
heartsuit	37
hline	45
hoffset	45
hsize	37
hslash	37
hspace	37
hspace*	37
htmladdnormallink	37, 45
htmlonly	54
htmlref	37, 45
huge	31
Huge	31
HUGE	31
hyphenation	46

## I

i	33
ieme	48
iemes	48
ier	48
iere	48
ieres	48
iers	48
iflatrixortf	46
ifx	46
iiint	37
iint	37
Im	37
in	37
include	37, 46
includegraphics	37
includegraphics*	37
includeonly	37
indent	37
index	37
indexentry	38
inferieura	48
infty	38
input	38, 46
int	38
iota	38
it	31, 54
itemize	54
itshape	31, 54

## J

j	33
---	----

## K

kappa	38
kern	33

## L

l	33
L	33
label	38
lambda	38
Lambda	38
land	38
landscape	54
langle	38
large	31
Large	31
LARGE	31
latex	33
LaTeX	33
LaTeXe	33
latexonly	54
latrixortffalse	46
latrixortftrue	46
lceil	38
LCS	48
ldots	38
le	38
left	38
leftarrow	38
Leftarrow	38
leftharpoondown	38
leftleftarrows	38
leftrightarrow	38
Leftrightarrow	38
leftrightarrows	38
leq	38
let	38
letter	54
letterspace	38
lfloor	38
lfoot	46
lhead	46
lim	38
liminf	38
limsup	38
linebreak	38
lineskip	38
link	53
list	55
listoffigures	38
listoffiles	46
listoftables	38
ll	38
longleftarrow	39
longleftarrows	39
longrightarrow	39
longtable	55
longtable*	55
lor	39
lower	33
lq	48
lqq	48
LyX	33

## M

makebox	39
makeglossary	46
makeindex	46
makelabels	46
maketitle	39
mapsto	39
marginpar	39
markboth	39, 46
markright	39, 46
math	55
mathbf	31
mathcal	31
mathit	31
mathmd	31
mathnormal	31
mathrm	31
mathsc	31
mathsf	31
mathsl	31
mathtt	32
mathup	32
matrix	39
mbox	39
mdseries	32, 55
measuredangle	39
medskip	39
mho	39
minipage	55
mit	32
moveleft	39
moveright	39
mp	39
mu	39
multicolumn	39, 55
music	55

## N

nabla	39
ne	39
nearrow	39
neg	39
neq	39
newblock	39
newcommand	46
newcount	39
newcounter	46
newenvironment	46
newfont	39
newif	46
newlength	46
newpage	39
newsavebox	39
newtheorem	46
nobibliography	39
nobreakspace	39, 46
nocite	39

nofiles	46
noindent	40
nolinebreak	40
nonfrenchspacing	40
nonumber	40
nopagebreak	40
normalfont	32
normalsize	32
notag	40
nu	40
numberline	40
numero	48
Número	48
numeros	48
Numeros	48
narrow	40

## O

oddsidemargin	46
og	48
omega	40
Omega	40
omicron	40
onecolumn	40
onlinecite	40
opening	47
oplus	40
oslash	40
otimes	40
output	40
overline	40

## P

pagebreak	40
pagenumbering	40, 46
pageref	40
Pageref	53
pagestyle	40, 46
par	40
paragraph	34
paragraph*	34
parbox	40
parindent	46
parskip	46
part	34
part*	34
partial	40
perp	40
phi	40
Phi	40
pi	40
Pi	40
picture	55
pm	41
pointexclamation	48
pointinterrogation	48

pointvirgule .....	48
possessivecite .....	52
prec .....	41
primo .....	48
printindex .....	41
prod .....	41
propto .....	41
protect .....	41
providecommand .....	46
ps .....	47
psfig .....	41
psfrag .....	41
psi .....	41
Psi .....	41

## Q

qquad .....	41
quad .....	41
quarto .....	48
quotation .....	55
quote .....	55

## R

r .....	33
raggedbottom .....	46
raggedright .....	31
raisebox .....	41
rangle .....	41
rawhtml .....	55
rceil .....	41
Re .....	41
ref .....	41
Ref .....	53
refstepcounter .....	41
renewcommand .....	46
renewenvironment .....	46
renewtheorem .....	46
resizebox .....	47
resizebox* .....	47
rfloor .....	41
rfoot .....	47
rhead .....	47
rho .....	41
right .....	41
rightarrow .....	41
Rightarrow .....	41
rightharpoonup .....	41
rightleftarrows .....	41
rightleftharpoons .....	41
rightrightarrow .....	41
rm .....	32, 55
rmfamily .....	32, 55
roman .....	41
Roman .....	41
rq .....	48
rqq .....	48

rule .....	41
------------	----

## S

S .....	53
samepage .....	41
savebox .....	41
sbox .....	41
sc .....	32, 55
scfamily .....	32
scriptsize .....	32
scshape .....	32, 55
searrow .....	41
section .....	34
section* .....	34
secundo .....	48
setbox .....	42
setcounter .....	47
setlength .....	47
settoheight .....	42
sf .....	32, 55
sffamily .....	32, 55
shortcite .....	52
shortciteA .....	52
shortciteauthor .....	52
shortciteNP .....	52
sigma .....	42
Sigma .....	42
signature .....	47
sim .....	42
simeq .....	42
sl .....	32, 55
SLiTeX .....	33
sloppypar .....	55
slshape .....	32, 55
small .....	32, 55
smallskip .....	42
spadesuit .....	42
sqrt .....	42
ssmall .....	32
stackrel .....	42
stepcounter .....	42
stretch .....	42
subparagraph .....	34
subparagraph* .....	34
subsection .....	34
subsection* .....	34
subset .....	42
subsetq .....	42
subsubsection .....	34
subsubsection* .....	34
succ .....	42
sum .....	42
superieura .....	48
supset .....	42
supseteq .....	42
surd .....	42
swarrow .....	42



## T

tabbing	55
table	55
table*	55
tableofcontents	42
tabular	55
tabular*	55
tau	42
tertio	48
TeX	33
textalpha	42
textbeta	42
textbf	32
textbullet	42
textchi	42
textcolor	42
textdelta	42
textDelta	42
textellipsis	42
textepsilon	42
texteta	42
textfont	32
textgamma	43
textGamma	43
textheight	47
textiota	43
textit	32
textkappa	43
textlambda	43
textLambda	43
textmd	32
textmu	43
textnormal	32
textnu	43
textomega	43
textOmega	43
textperiodcentered	43
textphi	43
textPhi	43
textpi	43
textPi	43
textpsi	43
textPsi	43
textrm	32
textsc	32
textsf	32
textsigma	43
textSigma	43
textsl	32
textsubscript	43
textsuperscript	43
texttau	43
textTau	43
texttheta	43
textTheta	43
texttt	32
textup	32
textwidth	47

textxi	43
textXi	43
textzeta	43
thanks	43
the	31
thebibliography	55
theendnotes	47
theindex	55
thepage	47
therefore	43
theta	43
Theta	43
thispagestyle	47
tilde	34
times	43
tiny	32
title	43
titlepage	55
to	43
today	31
topmargin	47
triangleleft	44
triangleright	44
tt	32, 55
ttfamily	32, 55
twocolumn	44
typeout	44
typein	44
Typein	44
typeout	44

## U

u	34
underbar	32
underline	32
up	48
uparrow	44
Uparrow	44
updownarrow	44
Updownarrow	44
upshape	32
upsilon	44
Upsilon	44
url	44
usebox	44
usepackage	47
uv	48

## V

v	34
value	44
varepsilon	44
varnothing	44
varphi	44
varpi	44
varpropto	44

<code>varsigma</code> .....	44
<code>vartheta</code> .....	44
<code>vbox</code> .....	44
<code>vcenter</code> .....	31
<code>vdots</code> .....	44
<code>vec</code> .....	34
<code>vee</code> .....	44
<code>verb</code> .....	44
<code>verb*</code> .....	44
<code>verbatim</code> .....	55
<code>Verbatim</code> .....	55
<code>verbositylevel</code> .....	47
<code>verse</code> .....	55
<code>voffset</code> .....	47
<code>vref</code> .....	44
<code>vsize</code> .....	44
<code>vskip</code> .....	44

<code>vspace</code> .....	44
<code>vspace*</code> .....	44

## W

<code>wedge</code> .....	44
<code>wp</code> .....	44

## X

<code>xi</code> .....	45
<code>Xi</code> .....	45
<code>xlink</code> .....	53

## Z

<code>zeta</code> .....	45
-------------------------	----

## 10 Concept Index

### %

`%latex2rtf` ..... 21

### A

`apacite` ..... 16

`apalike` ..... 16

`apanat1b` ..... 16

`authordate` ..... 16

### B

`babel` ..... 15

`bibliography` ..... 16

`bibliography, apacite` ..... 16

`bibliography, apalike` ..... 16

`bibliography, apanat1b` ..... 16

`bibliography, authordate` ..... 16

`bibliography, BibTeX` ..... 16

`bibliography, harvard` ..... 16

`bibliography, natbib` ..... 16

`BibTeX` ..... 16

### C

Conditional Parsing ..... 21

Copyright issues ..... 27

### D

`direct.cfg` ..... 21

`double spacing` ..... 17

### E

`equations` ..... 17

### F

`footnotes` ..... 16

### G

`geometry package` ..... 17

`graphics` ..... 18

### H

`harvard` ..... 16

History of the program ..... 27

`hyperlatex` ..... 19

### I

`iflaxtextortf` ..... 21

`ignore.cfg` ..... 21

`index` ..... 16

`input processing` ..... 21

`installation` ..... 3

### L

`language support` ..... 15

`line spacing` ..... 17

### M

`margins` ..... 17

`math` ..... 17

### N

`natbib` ..... 16

### P

`page formatting` ..... 17

`pagetypes` ..... 19

### R

`RTF sucks` ..... 1

### S

`setspace package` ..... 17

`special symbols` ..... 17

### T

`tables` ..... 18



# Table of Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Installation</b>	<b>3</b>
2.1	General	3
2.2	Obtaining L <sup>A</sup> T <sub>E</sub> X2RTF	3
2.3	UNIX	3
2.4	DOS	4
2.5	Win32 systems	5
2.6	Macintosh	6
2.7	Problems Compiling	6
2.8	Problems with <code>make check</code>	7
<b>3</b>	<b>Using L<sup>A</sup>T<sub>E</sub>X2RTF</b>	<b>9</b>
3.1	General Assumptions	9
3.2	L <sup>A</sup> T <sub>E</sub> X2RTF Options	9
3.3	Debugging	13
<b>4</b>	<b>Features</b>	<b>15</b>
4.1	L <sup>A</sup> T <sub>E</sub> X2e	15
4.2	Unicode Support	15
4.3	Input Encoding	15
4.4	Language Support	15
4.5	Cross References	16
4.6	Page Formatting	17
4.7	Equations	17
4.8	Math and Special Symbols	17
4.9	Tables	18
4.10	Graphics	18
4.11	Pagestyles	19
4.12	Hyperlatex	19
<b>5</b>	<b>Configuration</b>	<b>21</b>
5.1	Input processing	21
5.2	Conditional Parsing	21
5.3	Output Formatting	22
5.4	Direct Conversion	22
5.5	Ignore Command	22
5.6	Font Configuration	23
5.7	Language Configuration	23
<b>6</b>	<b>Error Messages and Logging</b>	<b>25</b>

<b>7</b>	<b>History &amp; Copyright .....</b>	<b>27</b>
<b>8</b>	<b>LaTeX2RTF under Development .....</b>	<b>29</b>
8.1	Unimplemented Features .....	29
8.2	Missing options .....	29
8.3	Known Bugs .....	29
8.4	Reporting Bugs .....	29
8.5	Todo List .....	30
8.6	Command List .....	30
8.6.1	General Commands .....	30
8.6.1.1	Basic Commands .....	30
8.6.1.2	Font Commands .....	31
8.6.1.3	Logos .....	33
8.6.1.4	Special Characters .....	33
8.6.1.5	Sectioning Commands .....	34
8.6.1.6	Uncategorized .....	34
8.6.2	Preamble Commands .....	45
8.6.3	Letter Commands .....	47
8.6.4	Language Commands .....	47
8.6.4.1	German Commands .....	47
8.6.4.2	Czech Commands .....	48
8.6.4.3	French Commands .....	48
8.6.4.4	Russian Commands .....	49
8.6.5	Citation Commands .....	50
8.6.5.1	Apacite Commands .....	51
8.6.5.2	AuthorDate Commands .....	52
8.6.5.3	Harvard Commands .....	52
8.6.5.4	HyperLatex Commands .....	53
8.6.5.5	Apacite Commands .....	53
8.6.6	Other Commands .....	53
8.6.7	Environments .....	54
<b>9</b>	<b>Function Index .....</b>	<b>57</b>
<b>10</b>	<b>Concept Index .....</b>	<b>65</b>