

L^AT_EX2RTF

A converter from L^AT_EX 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 Prah

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

$\text{\LaTeX}2\text{RTF}$ 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.

$\text{\LaTeX}2\text{RTF}$ 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. $\text{\LaTeX}2\text{RTF}$ 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 $\text{\LaTeX}2\text{RTF}$ — 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 27).

Some of the capabilities of $\text{\LaTeX}2\text{RTF}$ are restricted in scope or buggy (see [Section 8.3 \[Known Bugs\]](#), page 27).

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^AT_EX2RTF

L^AT_EX2RTF is available for many Unix Platforms, for the Macintosh, and for MS-DOS, including all versions of MS Windows.

The latest version of L^AT_EX2RTF 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 MS-DOS version (1.8aa and up) will also run under all MS Windows versions (including Windows Vista). It requires an i386 processor or better.

The Win32 distribution (starting from L^AT_EX2RTF 1.9.15) is the MS-DOS version plus Win32 GUI program (l2rshell).

There are a few people working on L^AT_EX2RTF, 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^AT_EX2RTF supports conversion of L^AT_EX equations to bitmaps using the shell script ‘`latex2png`’, found in ‘`scripts/`’. ‘`latex2png`’ requires that both L^AT_EX and ‘ImageMagick’ are installed. L^AT_EX2RTF 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 distributions do not contain an executable for DOS or Windows. You may compile it yourself (with `djgpp` or `MinGW`) or get the DOS distribution as file `'latex2rtf-x.x.x_dos.zip'` (where `x.x.x` is the version number) 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** (Win9x, WinME, WinNT, Win2000, WinXP, WinVista) **it is highly recommended to use the Windows version**. The Win32 version starting from 1.9.14a also has a Win32 GUI interface (`LATEX2RTF` Shell) and is wrapped in an automatic installer, see [Section 2.5 \[Win32 systems\]](#), page 5.

To install the DOS version of `LATEX2RTF`, extract all files from the zip archive, preserving the folder structure (winzip: check “use folder names”), preferably to `'C:\l2r'`, 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 $\text{\LaTeX}2\text{RTF}$ on a Win32 system (Win9x, WinME, WinNT, Win2000, WinXP, WinVista), download and execute the '`latex2rtf-x.x.x_win.exe`' (where x.x.x is the version number) and follow the instructions.

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

If your \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 before installing $\text{\LaTeX}2\text{RTF}$, the pathes to their executables are automatically found in the windows registry by the installer. Otherwise, these pathes may have to be specified manually in the "Environment" tab of the $\text{\LaTeX}2\text{RTF}$ GUI shell window.

If you need to start $\text{\LaTeX}2\text{RTF}$ 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 '`latex2rt`'.

Or you edit '`L2R.BAT`' and call '`L2R`' instead of calling '`LATEX2RT.EXE`'.

When equations are to be converted to bitmaps, for each equation `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 `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 `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 `LaTeX2RTF`. To convert a `LaTeX` file using this version, drag the file onto the `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 `LaTeX` and `'bibtex'`.

2.7 Problems Compiling

The code for `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 21). 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 20).

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 19).

See [Section 8.2 \[Missing options\]](#), page 27, for actual implementations irregularities.

See [Section 8.4 \[Reporting Bugs\]](#), page 27, 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 \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 21).

-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 $\text{\LaTeX}2\text{RTF}$ is supposed to translate correctly to RTF.

4.1 $\text{\LaTeX}2\text{e}$

$\text{\LaTeX}2\text{RTF}$ understands most of the commands introduced with $\text{\LaTeX}2_{\epsilon}$. 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, $\text{\LaTeX}2\text{RTF}$ has limited unicode support. \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 \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]{...}`. $\text{\LaTeX}2\text{RTF}$ 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 \LaTeX during processing. For example, this ensures that the $\text{\LaTeX}2\text{RTF}$ 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 \LaTeX file contains `\usepackage[language]{babel}`.

Encountering the ‘german’ package or documentstyle option (by H. Partl of the Viena University) makes $\text{\LaTeX}2\text{RTF}$ behave like that: German Quotes, German Umlauts by “a, etc. . . This support is programmed directly into $\text{\LaTeX}2\text{RTF}$ 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 21](#), 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

\LaTeX 2RTF 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 7. 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, \LaTeX 2RTF uses the comma as separator. If \LaTeX 2RTF 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 \LaTeX 2RTF 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 \LaTeX equations to MathType is letting \LaTeX 2RTF write the \LaTeX code of the equations verbatim in the `rtf` file (option `-M32`), then open the `rtf` file in Word, select the \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 20. 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 approach since the optional parameter is thrown away right now, see [Chapter 8 \[LaTeX2RTF under Development\], page 27](#).

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 13).

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 \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 \LaTeX followed by comma and a font name in RTF. The end is marked by a ‘.’. No spaces are allowed. The \LaTeX font will be converted to the RTF font when it is found in the \LaTeX file. If multiple translations for the same \LaTeX font are specified, only the first is used. All fonts in a \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 \LaTeX font — for example a Symbol font used in ‘`direct.cfg`’ — use a dummy \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 \LaTeX ’s “hardcoded” sectioning names. The standard \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 $\LaTeX 2_{\text{RTF}}$.

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 27).

‘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 27.

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 \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/p/latex2rtf/bugs/). 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 \LaTeX file along with a description of the problem. Isolating the bug into a small \LaTeX file does two things. First, it provides a file that can be used to test future versions of \LaTeX2RTF and second, it certainly improves the chances that the bug will get some attention. Do not send me large \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 \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 \LaTeX commands currently parsed by \LaTeX2RTF . Note: inclusion in this list does not mean that a command is fully and correctly handled by \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 \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; $\LaTeX 2_{RTF}$ 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_d

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

\		BCHAIR	49
\	31	BCHAIRS	49
		BCnt	49
		BCntIP	49
		because	33
		BEd	49
		BED	49
		BEDS	49
		begin	28
		Bem	49
		beta	33
		Beta	33
		bf	29, 52
		bfseries	29, 52
		bibentry	33
		bibitem	33
		bibliography	33
		bibliographystyle	33
		bibpunct	51
		BibTeX	31
		bigskip	33
		BIP	49
		BMTh	49
		BNUM	49
		BNUMS	49
		bot	33
		BOthers	49
		BOWP	49
		BoxedEPSF	33
		BPG	49
		BPGS	49
		BPhD	49
		BREPR	50
		breve	31
		BTR	50
		BTRANS	50
		BTRANSS	50
		bullet	33
		BUMTh	50
		BUPhD	50
		BVOL	50
		BVOLS	50
		C	
		c	31
		cal	29
		cap	33
		caption	33
		cc	45
		cdot	33
		cdots	33
		celsius	43
		center	52
		centering	33
A			
abstract	32, 52		
acknowledgments	52		
acute	31		
addcontents	32		
addcontentsline	32		
address	45		
addtocounter	43		
addtolength	43		
addvspace	32		
aleph	32		
align	52		
align*	52		
alltt	52		
alph	32		
Alph	32		
alpha	32		
Alpha	32		
amalg	32		
AmSLaTeX	31		
AmSTeX	31		
and	32		
angle	32		
appendix	33		
approx	33		
arabic	33		
array	52		
ast	33		
author	33		
AX	49		
B			
b	31		
BAnd	49		
bar	31		
baselineskip	33, 43		
BBA	49		
BBAA	49		
BBAB	49		
BBAY	49		
BBC	49		
BBCP	49		
BBCQ	49		
BBN	49		
BBOP	49		
BBOQ	49		
BCAY	49		
BCBL	49		
BCBT	49		

centerline	28	CYRCH	47
cfoot	43	cyrchsh	47
chapter	32	CYRCHSH	47
chapter*	32	cyrd	47
char	33	CYRD	47
chead	43	cyre	47
check	31	CYRE	47
chi	33	cyrerev	47
Chi	33	CYREREV	47
circ	33	cyrery	47
cite	33, 50, 51	CYRERY	47
Cite	51	cyrf	47
citeA	50	CYRF	47
citeaffixed	50	cyrg	47
citealp	51	CYRG	47
Citealp	51	cyrh	47
citealp*	51	CYRH	47
citealt	51	cyrhrdsn	47
Citealt	51	CYRHRDSN	47
citealt*	51	cyri	47
citeasnoun	50	CYRI	47
citeauthor	50, 51	cyrishrt	47
Citeauthor	51	CYRISHRT	47
citeauthor*	51	cyrk	47
citename	50	CYRK	47
citeNP	50	cyrl	48
citeonline	33	CYRL	47
citep	51	cyrn	48
Citep	51	CYRN	48
citep*	51	cyro	48
citet	51	CYRO	48
Citet	51	cyrp	48
citet*	51	CYRP	48
citetext	51	cyrre	48
citeyear	50, 51	CYRR	48
citeyear*	50	cyrs	48
citeyearNP	50	CYRS	48
citeyearpar	51	cyrsftsn	48
ck	45	CYRSFTSN	48
cleardoublepage	33	cyrsh	48
clearpage	33	CYRSH	48
cline	43	cyrt	48
closing	45	CYRT	48
clubsuit	33	cyru	48
comment	52	CYRU	48
compactenum	52	cyrv	48
compactitem	52	CYRV	48
cong	33	cyrya	48
contentsline	34	CYRYA	48
coprod	34	cyryu	48
cup	34	CYRYU	48
cyra	47	cyrz	48
CYRA	47	CYRZ	48
cyrb	47	cyrzh	48
CYRB	47	CYRZH	48
cyrc	47		
CYRC	47		
cyrch	47		

D

d	31
date	34
ddot	31
ddots	34
DeclareRobustCommand	43
DeclareRobustCommand*	43
def	43
degrecelsius	43
delta	34
Delta	34
description	52
deuxpoints	46
dfrac	34
Diamond	34
diamondsuit	34
displaymath	52
dittomark	46
div	34
document	52
documentclass	43
documentstyle	43
dot	31
doteq	34
dotfill	34
dots	34
doublespacing	43
downarrow	34
Downarrow	34

E

efloatseparator	34
ell	34
em	29, 52
emph	29
emptyset	34
encl	45
end	29
endinput	34
endnote	29
endnotemark	34
endnotetext	43
enotesize	29
ensuremath	34
enumerate	52
epsfbox	34
epsffile	34
epsilon	34
eqnarray	52
eqnarray*	52
eqref	34
equation	52
equation*	52
equiv	34
eta	34
EUR	43
euro	43

evensidemargin	43
exists	34

F

fancyfoot	43
fancyhead	43
fbox	34
FCS	46
fg	46
figure	52
figure*	52
flushbottom	43
flushleft	52
flushright	52
fnsymbol	34
footnote	29
footnotemark	34
footnotesize	29
footnotetext	43
forall	35
frac	35
Frac	35
framebox	35
frenchspacing	35
fullcite	50
fullciteA	50
fullciteauthor	50
fullciteNP	50
fup	46

G

gamma	35
Gamma	35
ge	35
geometry	43
geq	35
gg	35
glossary	35
glossaryentry	35
glq	45
glqq	45
grave	31
grq	45
grqq	45

H

H	31
harvardand	50
harvarditem	50
harvardyearleft	50
harvardyearright	50
hat	31
hbar	35
hbox	35
headheight	43

headsep.....	43
heartsuit	35
hline.....	43
hoffset.....	43
hsize.....	35
hslash.....	35
hspace.....	35
hspace*.....	35
htmladdnormallink	35, 43
htmlonly.....	52
htmlref	35, 43
huge.....	29
Huge.....	29
HUGE.....	29
hyphenation	44

I

i.....	31
ieme.....	46
imes.....	46
ier.....	46
iere.....	46
ieres.....	46
iers.....	46
iflatrixortf	44
ifx.....	44
iiint.....	35
iint.....	35
Im.....	35
in.....	35
include	35, 44
includegraphics	35
includegraphics*.....	35
includeonly	35
indent.....	35
index.....	35
indexentry	36
inferieura	46
infty.....	36
input	36, 44
int.....	36
iota.....	36
it.....	29, 52
itemize.....	52
itshape	29, 52

J

j.....	31
--------	----

K

kappa.....	36
kern.....	31

L

l.....	31
L.....	31
label.....	36
lambda.....	36
Lambda.....	36
land.....	36
landscape	52
langle.....	36
large.....	29
Large.....	29
LARGE.....	29
latex.....	31
LaTeX.....	31
LaTeXe.....	31
latexonly	52
latextortffalse.....	44
latextortftrue.....	44
lceil.....	36
LCS.....	46
ldots.....	36
le.....	36
left.....	36
leftarrow	36
Leftarrow	36
leftharpoondown.....	36
leftleftarrows	36
leftrightarrow	36
Leftrightarrow	36
leftrightarrows	36
leq.....	36
let.....	36
letter.....	52
letterspace	36
lfloor.....	36
lfoot.....	44
lhead.....	44
lim.....	36
liminf.....	36
limsup.....	36
linebreak	36
lineskip.....	36
link.....	51
list.....	53
listoffigures	36
listoffiles	44
listoftables	36
ll.....	36
longleftarrow	37
longleftarrows.....	37
longrightarrow	37
longtable	53
longtable*	53
lor.....	37
lower.....	31
lq.....	46
lqq.....	46
LyX.....	31

M

makebox	37
makeglossary	44
makeindex	44
makelabels	44
maketitle	37
mapsto	37
marginpar	37
markboth	37, 44
markright	37, 44
math	53
mathbf	29
mathcal	29
mathit	29
mathmd	29
mathnormal	29
mathrm	29
mathsc	29
mathsf	29
mathsl	29
mathtt	30
mathup	30
matrix	37
mbox	37
mdseries	30, 53
measuredangle	37
medskip	37
mho	37
minipage	53
mit	30
moveleft	37
moveright	37
mp	37
mu	37
multicolumn	37, 53
music	53

N

nabla	37
ne	37
nearrow	37
neg	37
neq	37
newblock	37
newcommand	44
newcount	37
newcounter	44
newenvironment	44
newfont	37
newif	44
newlength	44
newpage	37
newsavebox	37
newtheorem	44
nobibliography	37
nobreakspace	37, 44
nocite	37

nofiles	44
noindent	38
nolinebreak	38
nonfrenchspacing	38
nonumber	38
nopagebreak	38
normalfont	30
normalsize	30
notag	38
nu	38
numberline	38
numero	46
Número	46
numeros	46
Numeros	46
narrow	38

O

oddsidemargin	44
og	46
omega	38
Omega	38
omicron	38
onecolumn	38
onlinecite	38
opening	45
oplus	38
oslash	38
otimes	38
output	38
overline	38

P

pagebreak	38
pagenumbering	38, 44
pageref	38
Pageref	51
pagestyle	38, 44
par	38
paragraph	32
paragraph*	32
parbox	38
parindent	44
parskip	44
part	32
part*	32
partial	38
perp	38
phi	38
Phi	38
pi	38
Pi	38
picture	53
pm	39
pointexclamation	46
pointinterrogation	46

pointvirgule	46
possessivecite	50
prec	39
primo	46
printindex	39
prod	39
propto	39
protect	39
providecommand	44
ps	45
psfig	39
psfrag	39
psi	39
Psi	39

Q

qquad	39
quad	39
quarto	46
quotation	53
quote	53

R

r	31
raggedbottom	44
raggedright	29
raisebox	39
rangle	39
rawhtml	53
rceil	39
Re	39
ref	39
Ref	51
refstepcounter	39
renewcommand	44
renewenvironment	44
renewtheorem	44
resizebox	45
resizebox*	45
rfloor	39
rfoot	45
rhead	45
rho	39
right	39
rightarrow	39
Rightarrow	39
righttharpoonup	39
rightleftarrows	39
rightleftharpoons	39
rightrightarrow	39
rm	30, 53
rmfamily	30, 53
roman	39
Roman	39
rq	46
rqq	46

rule	39
------------	----

S

S	51
samepage	39
savebox	39
sbox	39
sc	30, 53
scfamily	30
scriptsize	30
scshape	30, 53
searrow	39
section	32
section*	32
secundo	46
setbox	40
setcounter	45
setlength	45
settoheight	40
sf	30, 53
sffamily	30, 53
shortcite	50
shortciteA	50
shortciteauthor	50
shortciteNP	50
sigma	40
Sigma	40
signature	45
sim	40
simeq	40
sl	30, 53
SLiTeX	31
sloppypar	53
slshape	30, 53
small	30, 53
smallskip	40
spadesuit	40
sqrt	40
ssmall	30
stackrel	40
stepcounter	40
stretch	40
subparagraph	32
subparagraph*	32
subsection	32
subsection*	32
subset	40
subsetq	40
subsubsection	32
subsubsection*	32
succ	40
sum	40
superieura	46
supset	40
supseteq	40
surd	40
swarrow	40

T

tabbing.....	53
table.....	53
table*.....	53
tableofcontents.....	40
tabular.....	53
tabular*.....	53
tau.....	40
tertio.....	46
TeX.....	31
textalpha.....	40
textbeta.....	40
textbf.....	30
textbullet.....	40
textchi.....	40
textcolor.....	40
textdelta.....	40
textDelta.....	40
textellipsis.....	40
textepsilon.....	40
texteta.....	40
textfont.....	30
textgamma.....	41
textGamma.....	41
textheight.....	45
textiota.....	41
textit.....	30
textkappa.....	41
textlambda.....	41
textLambda.....	41
textmd.....	30
textmu.....	41
textnormal.....	30
textnu.....	41
textomega.....	41
textOmega.....	41
textperiodcentered.....	41
textphi.....	41
textPhi.....	41
textpi.....	41
textPi.....	41
textpsi.....	41
textPsi.....	41
textrm.....	30
textsc.....	30
textsf.....	30
textsigma.....	41
textSigma.....	41
textsl.....	30
textsubscript.....	41
textsuperscript.....	41
texttau.....	41
textTau.....	41
texttheta.....	41
textTheta.....	41
texttt.....	30
textup.....	30
textwidth.....	45

textxi.....	41
textXi.....	41
textzeta.....	41
thanks.....	41
the.....	29
thebibliography.....	53
theendnotes.....	45
theindex.....	53
thepage.....	45
therefore.....	41
theta.....	41
Theta.....	41
thispagestyle.....	45
tilde.....	32
times.....	41
tiny.....	30
title.....	41
titlepage.....	53
to.....	41
today.....	29
topmargin.....	45
triangleleft.....	42
triangleright.....	42
tt.....	30, 53
ttfamily.....	30, 53
twocolumn.....	42
typeout.....	42
typein.....	42
Typein.....	42
typeout.....	42

U

u.....	32
underbar.....	30
underline.....	30
up.....	46
uparrow.....	42
Uparrow.....	42
updownarrow.....	42
Updownarrow.....	42
upshape.....	30
upsilon.....	42
Upsilon.....	42
url.....	42
usebox.....	42
usepackage.....	45
uv.....	46

V

v.....	32
value.....	42
varepsilon.....	42
varnothing.....	42
varphi.....	42
varpi.....	42
varpropto.....	42

<code>varsigma</code>	42
<code>vartheta</code>	42
<code>vbox</code>	42
<code>vcenter</code>	29
<code>vdots</code>	42
<code>vec</code>	32
<code>vee</code>	42
<code>verb</code>	42
<code>verb*</code>	42
<code>verbatim</code>	53
<code>Verbatim</code>	53
<code>verbositylevel</code>	45
<code>verse</code>	53
<code>voffset</code>	45
<code>vref</code>	42
<code>vsize</code>	42
<code>vskip</code>	42

<code>vspace</code>	42
<code>vspace*</code>	42

W

<code>wedge</code>	42
<code>wp</code>	42

X

<code>xi</code>	43
<code>Xi</code>	43
<code>xlink</code>	51

Z

<code>zeta</code>	43
-------------------------	----

10 Concept Index

%

`%latex2rtf` 19

A

`apacite` 14

`apalike` 14

`apanat1b` 14

`authodate` 14

B

`babel` 13

`bibliography` 14

`bibliography, apacite` 14

`bibliography, apalike` 14

`bibliography, apanat1b` 14

`bibliography, authodate` 14

`bibliography, BibTeX` 14

`bibliography, harvard` 14

`bibliography, natbib` 14

`BibTeX` 14

C

`Conditional Parsing` 19

`Copyright issues` 25

D

`direct.cfg` 19

`double spacing` 15

E

`equations` 15

F

`footnotes` 14

G

`geometry package` 15

`graphics` 16

H

`harvard` 14

`History of the program` 25

`hyperlatex` 17

I

`iflaxtextortf` 19

`ignore.cfg` 19

`index` 14

`input processing` 19

`installation` 3

L

`language support` 13

`line spacing` 15

M

`margins` 15

`math` 15

N

`natbib` 14

P

`page formatting` 15

`pagetypes` 17

R

`RTF sucks` 1

S

`setspace package` 15

`special symbols` 15

T

`tables` 16

Table of Contents

1	Introduction	1
2	Installation	3
2.1	General	3
2.2	Obtaining L ^A T _E 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>	6
3	Using L^AT_EX2RTF	7
3.1	General Assumptions	7
3.2	L ^A T _E X2RTF Options	7
3.3	Debugging	11
4	Features	13
4.1	L ^A T _E X2e	13
4.2	Unicode Support	13
4.3	Input Encoding	13
4.4	Language Support	13
4.5	Cross References	14
4.6	Page Formatting	15
4.7	Equations	15
4.8	Math and Special Symbols	15
4.9	Tables	16
4.10	Graphics	16
4.11	Pagestyles	17
4.12	Hyperlatex	17
5	Configuration	19
5.1	Input processing	19
5.2	Conditional Parsing	19
5.3	Output Formatting	20
5.4	Direct Conversion	20
5.5	Ignore Command	20
5.6	Font Configuration	21
5.7	Language Configuration	21
6	Error Messages and Logging	23

7	History & Copyright	25
8	LaTeX2RTF under Development	27
8.1	Unimplemented Features	27
8.2	Missing options	27
8.3	Known Bugs	27
8.4	Reporting Bugs	27
8.5	Todo List	28
8.6	Command List	28
8.6.1	General Commands	28
8.6.1.1	Basic Commands	28
8.6.1.2	Font Commands	29
8.6.1.3	Logos	31
8.6.1.4	Special Characters	31
8.6.1.5	Sectioning Commands	32
8.6.1.6	Uncategorized	32
8.6.2	Preamble Commands	43
8.6.3	Letter Commands	45
8.6.4	Language Commands	45
8.6.4.1	German Commands	45
8.6.4.2	Czech Commands	46
8.6.4.3	French Commands	46
8.6.4.4	Russian Commands	47
8.6.5	Citation Commands	48
8.6.5.1	Apacite Commands	49
8.6.5.2	AuthorDate Commands	50
8.6.5.3	Harvard Commands	50
8.6.5.4	HyperLatex Commands	51
8.6.5.5	Apacite Commands	51
8.6.6	Other Commands	51
8.6.7	Environments	52
9	Function Index	55
10	Concept Index	63