Manual for Tex2RTF 1.0: a LaTeX to RTF and HTML converter

Julian Smart
Artificial Intelligence Applications Institute
University of Edinburgh
EH1 1HN

October 1993

Contents

Copyright notice

Introduction

This document describes a utility for converting LaTeX files into several other formats, and the C++ library which the converters rely on for parsing and other services.

Only a subset of LaTeX can be processed by this utility, especially since the target document language will never perfectly match LaTeX. Whether the quality of the results are good enough will depend upon the application and your own expectations.

Tex2RTF is heavily biassed towards making on-line, hypertext versions ofLaTeX documents, but the RTF converter can be used for normal, linear documents too.

The latest version of Tex2RTF can be accessed by anonymous ftp from skye.aiai.ed.ac.uk (192.41.104.6) in the directory /pub/tex2rtf. It is available in SPARC Open Look and Windows 3.1 versions.

Tex2RTF was developed using the free Open Look/Motif/Windows 3.1 C++ class library wxWindows, also available from the above FTP site in the /pub/wxwin/beta directory.

Hypertext features

LaTeX is inherently suitable for specifying hypertext documents since it encourages description of the logical structure of a document using section commands. Therefore, a LaTeX document is automatically a hypertext document, without any further editing.

For Windows Help, a single RTF is generated with topics corresponding to sections. A top level contents page shows each chapter or top-level section, and each chapter or section ends with a list of further sections or subsections. Similarly, a single wxHelp XLP file may be generated.

For HTML, a different file is generated for each section, since the XMOSAIC browser works best with a large number of small files. The files are named automatically based on the name of the output file, with the contents page filename being formed from the output filename with `_contents` appended to the name. This may result in the generation of several hundred files for a large LaTeX input file.

To specify explicit jumps around a hypertext file, the helpref macro is used. The first argument is the text to be displayed at the point of reference, which will be highlighted in a hypertext file to allow jumping to a reference. The second argument is the reference label (there should be a corresponding label command in the file, following a section or figure).

A first optional argument (written in square brackets) specifies a filename which contains the given reference. (Not yet implemented.)

To use extra features such as **helpref** and the C++ and CLIPS class reference documentation features, include the style file `texhelp.sty`.

Running Tex2RTF

Tex2RTF accepts two arguments (input and output filenames) and trailing (optional) switches. If both filenames are given, the utility will work in batch mode. Otherwise, a main window will be shown, will appropriate menu items for selecting input and output filenames, starting off the conversion process, and so on.

It is recommended that you run Tex2RTF twice in order to be sure of of resolving all references and including an up-to-date contents page.

-interactive Forces interactive mode even if both filenames are given.

-macros filename Specifies a file for the custom macro file -- see Macro not found error.

-winhelp Specifies Windows Help RTF output.

-rtf Specifies linear RTF output.

-html Specifies HTML (World Wide Web) output.

-twice Tells Tex2RTF to run the conversion twice to ensure all references and citations are resolved and the contents page included.

Macro reference

The following lists macros which are recognised by the converters. Other macros not not mentioned can be assumed to be unrecognised or ignored. Each macro is given with its number of arguments (not including any optional arguments).

abstract:1
addcontentsline:3
author:1
backslash:0
bf:1
bibitem:2
bibliographystyle:1
bibliography:0
boxit:1
caption:1
cdots:0
centerline:1
center:1
cextract:0
chapter*:1
chapter:1
cinsert:0
cite:1
class:1
clipsfunc:3
comment:1
copyright:0
cparam:2
date:1
description:1
documentstyle:1
document:1
em:1
enumerate:1
fbox:1
figure:1
flushleft:1
flushright:1
footnote:1
framebox:1
functionsection:1
func:3
helpignore:1
helponly:1
helpinput:1
helpfontfamily:1
helpfontsize:1
helpref:2
helprefn:2
hline:0
hrule:0
huge:1
Huge:1
HUGE:1

include:1
input:1
itemize:1
item:0
image:2
it:1
label:1
large:1
Large:1
LARGE:1
LaTeX:0
ldots
maketitle
mbox:1
membersection:1
member:1
newcommand:3
newpage:0
nocite:1
noindent:0
normalsize:1
param:1
parindent:1
parskip:1
par:0
psboxto:2
quote:1
quotation:1
ref:1
rm:1
rtfsp:0
sc:1
section*:1
section:1
shortcite:1
small:1
subsection*:1
subsection:1
subsubsection*:1
subsubsection:1
tabbing:1
tableofcontents:0
tabular:2
TeX:0
title:1
tiny:1
today:0
tt:1
typeout:1
underline:1
verbatiminput:1
verbatim:1

Errors and bugs

Errors
Bugs

References

[1] **Smart, Julian**. 1993. *wxWindows 1.50 User Manual.* University of Edinburgh. Artificial Intelligence Applications Institute. 80 South Bridge, Edinburgh, EH1 1HN.

[2] **Akagi, Shinsuke and Fujita, Kikuo**. 1987. Building an expert system for the preliminary design of ships. *AI EDAM*, **1**(3), pages 191-205.

[3] **Allen, J. and others**. 1987. *From text to speech: the MITalk System.* Camb. Univ. Press. New York.

[4] **Bruce, Bertram C.** 1982. Natural communication between person and computer, from *Strategies for Natural Language Processing.* Lawrence Erlbaum Associates. Chap. 3, pages 16-87.

[5] **Benyon, P.R.** 1988. Presenting models in plain English, from *Mathematics and Computers in Simulation.* Number 30. Pages 17-25.

[6] **Drabble, Brian**. 1988. *Intelligent execution monitoring and error analysis in planning involving processes.* PhD thesis. University of Aston. University of Aston. Birmingham.

[7] **Forbus, Kenneth and Stevens, Albert**. 1981 (March). *Using qualitative simulation to generate explanations: report no. 4490.* Technical report. Navy Personnel Research and Development Center.

[8] **Allen, James**. 1987. *Natural Language Understanding.* Benjamin/Cummings.

Why use LaTeX?

LaTeX happens to be a very convenient format if you need to produce documents (such as manuals, help facilities, up-to-date information) in both printed and on-line media. Being a language rather than a WYSIWYG system, it allows explicit specification of layout and document structure, lending itself well to hypertext applications and automatic document generation. Many people also prefer to use LaTeX for ordinary use since it encourages a logical document structure and the user is not distracted by having to perfect the appearance; many layout decisions are taken by LaTeX automatically.

Although LaTeX is not as fancy as modern word processors and desk-top publishing packages, it is for many purposes quite adequate, and sometimes more flexible than its modern counterparts.

The conversion utility gives LaTeX a new lease of life by allowing virtually all other wordprocessor formats to be generated from documents containing a reasonable subset of LaTeX syntax. From the same LaTeX sources, we can now generate printed manuals, Windows Help files, wxHelp files, RTF-compatible word processors formats such as MS Word, and HTML files for use in the World Wide Web. Since the conversion tool is free, as are LaTeX, HTML viewers, wxHelp and (effectively) Windows Help, there are no financial or time penalties for providing documentation in a wide range of printed and hypertext formats.

Output Formats

At present the following output formats are supported:

- RTF (Rich Text Format). This is the most well-developed converter. RTF is commonly used as a document exchange format amongst Windows-based applications, and is the input for the Windows Help Compiler. Tex2RTF supports both linear documents and Windows Help hypertext format.

- wxHelp. This is the platform-independent help system for the class library wxWindows [1]. It can display ASCII files with embedded codes for changing font styles, but no formatting is done by wxHelp.

- HTML (Hypertext Markup Language). This an SGML-like format commonly used by documents in the World Wide Web distributed hypertext system, and formats text dynamically rather like Windows Help.

What compromises must I make?

As a LaTeX user, you need to be aware that some macros or facilities don't transfer to other formats, either because they are not supported by the target format or because the converter does not support them. Example of this are the footnote macro, and all maths formatting.

Sometimes LaTeX facilities must be accessed in a slightly different way to support the variety of formats, particularly hypertext formats where LaTeX references are often replaced by hypertext jumps (but must still look right in printed documentation). Tables don't transfer well to RTF (and not at all to the other formats) but an attempt is made to approximate tables so long as special row macros are used, instead of the usual end of row delimiter.

Bibliographies are handled quite well since the utilities can read in `.bib` files and resolve citations. Bibliographies are not yet supported in Windows Help files, and the references are not yet sorted alphabetically. Numbers are used in citations.

Pictures are handled in a limited way: if the PSBOX macro package is used, an *image* macro can be used to place Encapsulated PostScript files in LaTeX, and Windows RGB-encoded bitmap files when converting to RTF.

Paragraph indentation is a little weak at present, but some is done automatically, in list and quotation environments. More flexible control will be added later.

Nested file inclusion (input, include, verbatiminput), is handled, and the comment environment is supported. However, using *input* to include macro packes is not advisable, although the psbox.tex package is ignored. In later versions, it should be possible to specify files to ignore.

Because of the way LaTeX is parsed, some syntax has to conform to a few simple rules. Macros such as *bf* and *it* need to occur immediately after a left brace, and have a block of their own, since the text within their scope is regarded as its argument. This syntax means the same thing as using *begin ... end*, which is usually a one argument macro (the argument is the text between the *begin* and *end*). See Space.

As a Windows hypertext help writer, you don't have access to all RTF commands but you'll be able to get most of what you want. In particular, any LaTeX document you write will automatically be a hypertext document, because the converter takes advantage of the hierarchy of sections. Futher jumps can be placed using a combination of the label and helpref macros.

Similarly, HTML support is largely automatic, and multiple files are generated from one LaTeX file since browsing HTML works best with many small files rather than a few large ones.

wxHelp files are least well supported since there is no formatting support, only font style, sizes and colours. Still, some hypertext help support on UNIX/X platforms is better than none. The class library wxWindows may be extended in future to allow using a better help viewer, such as *xmosaic*. Of course there is nothing to stop xmosaic being used as a help system, but it won't be integrated with wxWindows programs as wxHelp is.

Sometimes you will use a local macro package that is unrecognised by the converters. In this case, you may define a custom macro file where macros are defined in terms of supported LaTeX commands and text. Even if the result is not the same as in LaTeX, you can probably end up with something adequate, and at least avoid undefined macro errors. See Errors for further information.

Changes to LaTeX syntax

Here are the conventions you need to observe to satisfy the Tex2RTF parser.

abstract:1

Writes the title **Abstract**. Linear RTF.

addcontentsline:3

Adds a title to the contents page. Linear RTF.

author:1

Defines the author, for output when *maketitle* is used.

backslash:0

Outputs a backslash in math mode (should be enclosed by two dollar symbols).

bf:1

Specifies bold font.

bibitem:2

For convenience, *bibitem* has two arguments: label and item. LaTeX syntax permits writing this as if it were two arguments, even though it is in fact only one.

bibliographystyle:1

Currently doesn't affect the style of bibliography, but probably will in the future.

bibliography:0

Includes the bibliography at this point in the document.

boxit:1

Draws a box around the paragraph.

caption:1

Specifies a caption (within a figure environment). This may be followed immediately by a label command.

cdots:0

Outputs three dots.

centerline:1

Centres (or centers!) a line of text.

center:1

Centres a block of text.

cextract:0

Prints a C++ extraction operator (>>).

chapter*:1

Outputs a chapter heading with no contents entry.

chapter:1

Outputs a chapter heading.

cinsert:0

Prints a C++ insertion operator (<<).

cite:1

Cite a reference. The argument is a reference key as defined in a LaTeX .bib file.

class:1

Outputs the argument, an index entry (LaTeX only) and a keyword entry (WinHelp only). Used in class reference documentation.

clipsfunc:3

Formats a CLIPS function, given the return value, function name, and arguments.

comment:1

Allows large comments in LaTeX files. Ignored by Tex2RTF.

copyright:0

Outputs the copyright symbol.

cparam:2

Formats a CLIPS type and argument. Used within the third argument of a clipsfunc command.

date:1

Specifies the date of a document; only output by maketitle.

description:1

A list environment, where each item command must be followed by optional (!) square-bracketed text which will be highlighted.

documentstyle:1

Specifies the main style (report, article etc.) and, optionally, style files such as `texhelp.sty`. A report has chapters, while an article's top-level sections are specified using section.

document:1

This environment should enclose the body of a document.

em:1

Emphasizes text (italic in RTF).

enumerate:1

Enumerate list environment: numbers the items.

fbox:1

Boxes a paragraph in LaTeX and RTF.

figure:1

A figure enviroment: does nothin in RTF.

flushleft:1

Flushes the given text to the left margin.

flushright:1

Flushes the given text to the right margin.

footnote:1

Footnotes are not yet implemented in RTF: the text is bracketed instead.

framebox:1

Boxes a paragraph in LaTeX and RTF.

functionsection:1

Defines a subsection, adding the C++ function name to the LaTeX index or the WinHelp keyword list.

Should be followed by a func command to specify function details.

func:3

Defines a C++ function, given the return type, function name, and parameter list.

Should occur after a functionsection command.

helpignore:1

Ignores the argument in Tex2RTF generated files, but not LaTeX.

helponly:1

Only outputs the argument in Tex2RTF generated files.

helpinput:1

Only includes the given file in Tex2RTF generated files.

helpfontfamily:1

Specifies the font family for Tex2RTF generated files. The argument may be Swiss or Times.

helpfontsize:1

Specifies the font size for Tex2RTF generated files.

helpref:2

Specifies a jump to a labelled chapter, section, subsection subsubsection or figure.

The first argument is text to be highlighted (mouseable in help systems) and the second is the reference label. In linear documents, the section number is given following the text, unless the helprefn command is used instead, where the section number is suppressed.

An optional first argument in square brackets specifies a filename where the topic is found (not yet implemented).

helprefn:2

Specifies a jump to a labelled chapter, section, subsection subsubsection or figure.

The first argument is text to be highlighted (mouseable in help systems) and the second is the reference label. See helpref for the form where the section number is printed in linear documents.

An optional first argument in square brackets specifies a filename where the topic is found (not yet implemented).

hline:0

Within a tabular environment, draws a horizontal rule below the current row.

hrule:0

Draws a horizontal line below the current paragraph.

huge:1

Outputs the argument in huge text.

Huge:1

Outputs the argument in huger text that huge.

HUGE:1

Outputs the argument in huger text that Huge.

include:1

Include the given file. The command must not be preceded by any whitespace, and spurious whitespace between elements of the command will also trip up Tex2RTF.

input:1

Include the given file. The command must not be preceded by any whitespace, and spurious whitespace between elements of the command will also trip up Tex2RTF.

itemize:1

Indents each item of a list and precedes with a bullet.

item:0

Marks an item of a itemize, description or enumeratelist. Items within a description environment should have an 'optional' argument in square brackets which will be highlighted.

image:2

This is translated to a PSBOX macro package *psboxto* command in LaTeX, the first argument being a sizing command and the second a filename.

In HTML mode, the second argument is used to generate a PostScript file reference.

In RTF mode, the second argument is used to find an RGB-encoded Windows bitmap file and insert it in the output file. The file *must* be RGB-encoded; RLE-encoded files don't seem to work. Note that this facility has not been rigorously tested.

it:1

Marks the argument in italic.

label:1

Labels the chapter, section, subsection, subsubsection or figure caption with the given label. This must be an ASCII string, and duplicate items with different case letters are not allowed.

The command must follow immediately after the section or caption command, with no intervening whitespace.

large:1

Marks the argument in large text.

Large:1

Makes the argument display in larger text than large.

LARGE:1

Makes the argument display in larger text than Large.

LaTeX:0

Outputs the annoying LaTeX upper and lower case name.

ldots

Outputs three dots.

maketitle

Makes the article or report title by outputting the title, author and optionally date.

mbox:1

Draws a box around the given paragraph in LaTeX and RTF, for example:

This should be a boxed paragraph for highlighting important information, such as information for registering a shareware program.

membersection:1

Used when formatting C++ classes to print a subsection for the member name.

member:1

Used to format a C++ member variable name.

newcommand:3

Define a new command; arguments are the command, the number of arguments, and the command body. For example:

```
\newcommand{\crazy}[2]{{\bf #1} is crazy but {\bf #2} is not.}
```

The command must have no whitespace at the start of the line or between the three arguments.

New commands may also be defined in the `tex2rtf.ini` file using slightly different syntax (see Macro not found error).

newpage:0

Inserts a page break.

nocite:1

Specifies that this reference should appear in the bibliography, but the citation should not appear in the text.

See also cite.

noindent:0

Sets paragraph indentation to zero. See also parindent.

normalsize:1

Sets the font size back to normal.

param:1

Formats a C++ type and argument pair. Should be used within the third argument of a a func command.

parindent:1

Indents the first line of succeeding paragraphs by the given amount.

parskip:1

Changes the spacing between paragraphs. In fact, in RTF this will cause two par commands to be output if parindent is greater than zero.

par:0

Causes the paragraph to end at this point. LaTeX and Tex2RTF also treat two consecutive newlines as a paragraph break.

psboxto:2

Identical to image.

quote:1

Indents a short quotation.

quotation:1

Indents a long quotation.

ref:1

In LaTeX and linear RTF, refers to a label and causes the number of that section or figure to be printed.

rm:1

Causes the argument to be formatted in a plain, roman font.

rtfsp:0

Outputs a space in RTF. Tex2RTF tries to insert a space where one is implied by a newline, but cannot cope where a line starts or ends with a command, in the middle of a paragraph. Use this command to insert a space explicitly.

sc:1

Prints the output in small capitals.

section*:1

Section header, with no entry in the contents page.

section:1

Section header, with an entry in the contents page.

shortcite:1

The same as cite.

small:1

Prints the argument in a small font.

subsection*:1

Subsection header, with no entry in the contents page.

subsection:1

Subsection header, with an entry in the contents page.

subsubsection*:1

Subsubsection header, with no entry in the contents page.

subsubsection:1

Subsubsection header, with an entry in the contents page.

tabbing:1

Tabbing environment: doesn't work properly in RTF.

tableofcontents:0

Inserts the table of contents at this point.

tabular:2

Tabular environment: an attempt is made to output something reasonable in RTF and XLP formats, although only simple tables will work.

TeX:0

Outputs the annoying TeX upper and lower case name.

title:1

Sets the title, to be output when the command maketitleis used.

tiny:1

Prints the argument in a very small font.

today:0

Outputs today's date.

tt:1

Outputs the argument in teletype font.

typeout:1

Outputs the text on the Tex2RTF text window.

underline:1

Underlines the argument.

verbatiminput:1

Include the given file as if it were within a verbatimenvironment. The command must not be preceded by any whitespace, and spurious whitespace between elements of the command will also trip up Tex2RTF.

verbatim:1

Uses a fixed-width font to format the argument without interpreting any LaTeX commands.

Errors

The following sections explain some of the errors that may occur, and how to deal with them.

Bugs

Command parsing. If a command is used followed by inappropriate argument syntax, Tex2RTF can crash. This an occur when a command is used in an asterisk form that is only formed in the non-asterisk variety. The non-asterisk form is assumed, which makes the following asterisk trip up the parser.

Bibliography. There's no flexibility in the way references are output: I expect I'll get round to doing something better, but only if people tell me they need it!

Tables. Tables aren't handled very well. Any suggestions how they could be done better?

Footnotes. Not yet supported, since I can't get MS Works to recognise them in RTF files.

Indexes and glossaries. Not yet supported.

Space

Tex2RTF attempts to insert spaces where LaTeX assumes whitespace. However, for the benefit of RTF conversion, you need to use the \ *rtfsp* macro where a command or brace within a paragraph begins or ends with a macro. For example:

```
Within a paragraph, you need to be careful about commands that begin
\rtfsp {\it at the start} of a line.
```

As normal with LaTeX, two newlines represents a paragraph break, although par can also be used at the end of a paragraph.

You need to have a blank line between section and some environment commands and the first paragraph or your document will look rather weird, e.g. headings running into paragraphs.

wxHelp is more fussy than LaTeX or RTF: you need to use percent characters at line ends liberally to eliminate newlines after commands on single lines.

Command arguments

All commands which have more than one argument can be used in the following three ways:

```
\begin{bf}
Some text.
\end{bf}

\bf{Some text.}

{\bf Some text.}
```

With the third method, it is important that the command has its own pair of braces, and that the command immediately follows the first brace. Otherwise, the parser cannot parse the argument(s) properly. With multiple arguments, each should be enclosed in braces.

Optional arguments are specified using square brackets or parentheses.

Whitespace should be avoided between command names and their arguments.

Labels

The *label* command may be used for sections and figure captions, but must come immediately after the section or caption commands with no intervening whitespace.

Macro not found

This error indicates that Tex2RTF has not implemented a standard LaTeX macro, or that a local macro package is being used that Tex2RTF does not know about. It can cause spurious secondary errors, such as not recognising the end document command.

You can get round this by defining a macro file (default name `tex2rtf.ini`) containing command definitions, such as:

```
\crazy      [2]{{\bf #2} is crazy but #1 is not}
\something  [0]{}
\julian     [0]{Julian Smart}
```

New commands may be defined in LaTeX files, but custom macro files will have to be defined when local style files are being used.

Unresolved references

References and citations are usually resolved on a second pass of Tex2RTF. If this doesn't work, then a missing label or bibliographical entry is to blame.