

Word
to
L^AT_EX

User's Manual

Michal Kebrt

Contents

1	User's manual	3
1.1	Requirements and installation	3
1.2	Uninstallation	4
1.3	Configuration	4
1.4	Command-line convertor	4
1.5	EPS to TIF image conversion	5
1.6	Graphic user interface	5
1.6.1	Running the conversion	5
1.6.2	Figures, Equations and Translations	6
1.6.3	Document preamble	8
1.6.4	Special characters	9
1.6.5	Styles and Font sizes	10
1.6.6	Miscellaneous options	11
1.7	Running Word-to-L ^A T _E X from Word	12
1.8	Conversion to XML, XHTML, MathML	12
A	Sample documents	14
B	Structure of configuration files	19
B.1	Conversion options	20
B.2	Conversion mappings	23
B.3	Special characters	37

Chapter 1

User's manual

1.1 Requirements and installation

- *Microsoft Windows 2000* or *XP* is required.
- *Microsoft .NET Framework Version 1.1* or higher is required. We strongly recommend *.NET Framework 1.1* because the convertor cannot be run as a Word addin with *.NET Framework 2.0*. Only the standalone version (which is much slower) can be run with *.NET Framework 2.0*. *.NET Framework 1.1* can be downloaded from Microsoft and it can be installed together with *.NET Framework 2.0* if you already have it.
- *Microsoft Word XP (2002)* or higher is required to be installed on your system.
- If you want to export mathematical equations not only as images, but also to \LaTeX or MathML formats, you will have to install *Design Science MathType* (it's a commercial product).
- You must have a PostScript printer driver installed on your system to be able to export images to EPS format. You can try this printer.

After you have installed all the required software, close *Word* (if it's running), execute `setup.exe` in the `setup\Word-to-LaTeX` directory, and follow the instructions. You must have administrator privileges to install the whole application properly. Once the installation is finished, you will find a couple of files in your *Word-to- \LaTeX* directory. Some of them are listed here:

- `word-to-latex.exe` – *Word-to- \LaTeX* command-line convertor
- `word-to-latex-gui.exe` – *Word-to- \LaTeX* graphic user interface
- `config.xml`, `XMLconfig.xml` – convertor configuration for \LaTeX and XML output
- `html.xsl` – XSL file which transforms XML output to HTML
- `manual.pdf` – user's manual
- `eps2tif` – directory containing a batch file for converting EPS images to TIF format

1.2 Uninstallation

If you want to uninstall *Word-to-L^AT_EX* from your system, go to **Control Panel | Add or Remove programs** and select *Word-to-L^AT_EX*. Please close *Word* (if it's running) before uninstalling.

1.3 Configuration

All the program configuration is stored in an XML file with a public format which is defined using XML Schema in the `config.xsd` file. Before the conversion procedure starts, the configuration is validated against the schema, so you must be very careful when editing the file manually.

There are two predefined configuration files in your *Word-to-L^AT_EX* directory, `config.xml` for conversion to L^AT_EX and `XMLConfig.xml` for conversion to XML format.

Don't be afraid if XML is an unknown abbreviation for you. There is no need to know anything about XML technologies because you can customize the convertor also through the graphic interface which will be described in section 1.6.

Appendix B describes the XML structure of configuration files and possible values in each element and attribute.

1.4 Command-line convertor

When the command-line convertor (`word-to-latex.exe`) is executed without any parameters, the list of all possible options from table 1.1 will be printed.

```
word-to-latex.exe -i inputFile [-o outputFile] [-opt configFile]
  -i | input file name
  -o | output file name
  -opt | configuration file name
```

Table 1.1: `word-to-latex.exe` options

The only required option is “-i”. When the output file is omitted, the input file name appended with “.tex” extension is taken instead. If the configuration file is not specified, the default configuration stored in the `config.xml` file is used for the conversion.

After you run the program with correct options, it prints all the file names (input, output, configuration) and also your *Microsoft Word* version which can be useful when an error occurs. Then the conversion routine is started and you will be informed about the progress.

Please be patient when you are converting a large document, it can take a long time to convert it. Much more faster way of running the conversion will be described in section 1.7.

1.5 EPS to TIF image conversion

As not all images included in *Word* documents can be converted to bitmaps, I wrote a simple batch file (`eps2tif.bat` in the `eps2tif` directory) which converts EPS files to TIF format. It benefits from the fact that *Word-to-L^AT_EX* can export all images to EPS format.

This batch file requires *Ghostscript* program which is free for non-commercial use. The path to the *Ghostscript* executable must be specified at the top of the `eps2tif.bat` file.

When you want to export all images from a *Word* document to some bitmap format (PNG, JPEG, and so on), just run *Word-to-L^AT_EX* to have an EPS version of each image and then execute the `eps2tif.bat` file with the options described in table 1.2. Finally you can convert the output TIF files to the format you prefer (for example *Irfanview* does this very effectively).

<code>eps2tif.bat inDir outDir</code>	
<code>inDir</code>	directory from which the files with <code>.eps</code> extension are taken
<code>outDir</code>	directory where the <code>.tif</code> files will be saved

Table 1.2: `eps2tif.bat` options

1.6 Graphic user interface

For most of users the graphic interface will be the most frequent way of using *Word-to-L^AT_EX* convertor. To run it, just click the icon on your Desktop or in the Start menu, or execute the `word-to-latex-gui.exe` file in your *Word-to-L^AT_EX* directory.

After executing the program, the configuration dialog will appear. All the six tabs will be described now.

1.6.1 Running the conversion

Only the **Input document** is required to be selected. When the **Output file** is omitted, the **Input document** file name appended with “`.tex`” extension is taken instead.

Two configuration files can be found in your *Word-to-L^AT_EX* directory, `config.xml` for conversion to L^AT_EX and `XMLConfig.xml` for conversion to XML.

When the **Configuration file** is omitted, `config.xml` will be used instead. But be careful, it’s recommended to customize the settings for each document you convert. **Save as ...**, **Save** and **Load** commands in the **Configuration** menu can be used to load and save convertor configurations. Remember that the current configuration must be saved before it is applied during the conversion. You can check the option **Save configuration before conversion** to save the configuration automatically after pressing the **Convert** button.

When you press the **Convert** button, all the file names (input, output, configuration) and also your *Microsoft Word* version will be written to the text box

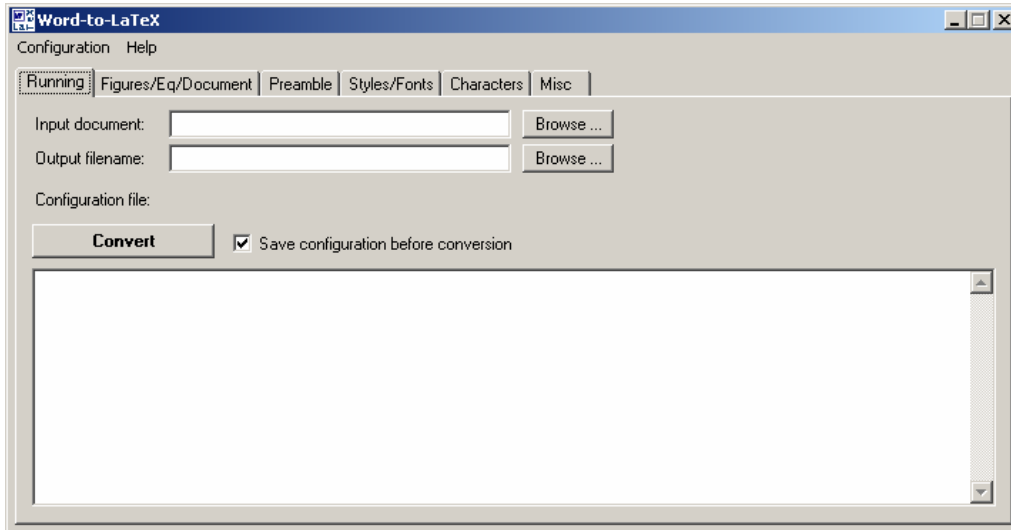


Figure 1.1: “Running” tab

below. This can be useful when an error occurs. Then the conversion routine is started and you will be informed about the progress in the text box. Please be patient when you are converting a large document, it can take a long time to convert it. Much more faster way of running the conversion will be described in section 1.7.

1.6.2 Figures, Equations and Translations

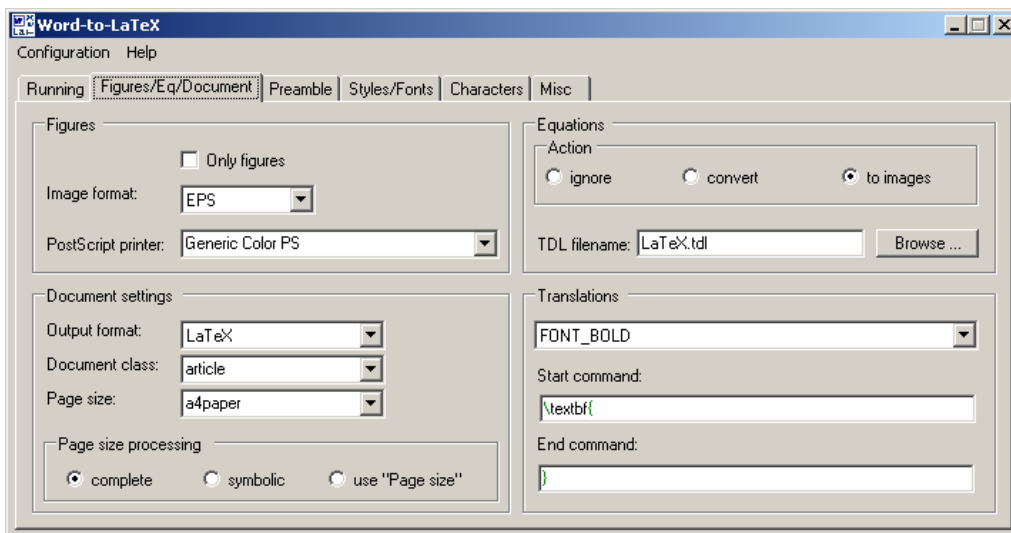


Figure 1.2: “Figures/Eq/Document” tab

Figures

Check **Only figures** to convert only figures and ignore the text content of the input document. *Word-to-L^AT_EX* exports images (including embedded objects like *Excel* graphs) in two formats – vector Encapsulated PostScript (**EPS**) or bitmap **PNG**. If you want to export images to EPS format, you must specify the **PostScript printer**. This topic was mentioned in section 1.1.

EPS format is recommended because EPS images can be easily integrated into L^AT_EX documents and moreover some images included in *Word* documents (e.g. *Word* drawings) cannot be exported as bitmaps. If this occurs, the convertor will give you a notice and after it finishes, you can export all images to EPS format and use `eps2tif` program described in section 1.5 to have a bitmap version of each image.

Equations

If you have *MathType* installed on your system, you can check **convert** and all equations inserted through *Equation Editor*, *MathType* and *Word* EQ fields will be converted. Otherwise you have to select **ignore** to ignore all equations or **to images** for exporting equations to images.

When the **convert** option is selected, the output format of converted equations depends on the translation file defined in the **TDL filename** box. See the **Translators** subdirectory of your *MathType* directory for possible values. You can edit or add new files to this directory if you want to customize the conversion of equations.

Document settings

As the convertor performs a few special actions depending on the **Output format**, you must select **L^AT_EX** or **XML**. But remember that it doesn't change any **Translations**.

The `@WL-DOC_CLASS` macro used in the document preamble will be replaced with the value of the **Document class** option. The `@WL-PAGE_SIZE` macro will be replaced with a value depending on the **Page size processing** option as shows table 1.3.

Option name	@WL-PAGE_SIZE will be replaced with
complete	the complete definition of the page size matching the page size of the input document
symbolic	the convertor will try to translate the symbolic page size (e.g. A4) of the input document to an appropriate L ^A T _E X size (e.g. letterpaper)
use “Page size”	the value of the Page size option

Table 1.3: **Page size processing** options

Translations

The translation mappings between input document elements and L^AT_EX commands are defined here. It comprises of headings, font styles, footnotes, tables,

alignments, colors, and so on. Each element has a **Start command** which is inserted before the element itself and an **End command** inserted after the element.

One example: Let “*some text*” appear in the document and the FONT_ITALIC mapping is “\textit{” for the start command and “}” for the end command. Then “\textit{Some text}” will be written to the output file.

The complete overview of translated elements with the default mappings for L^AT_EX and XML output can be found in section B.2.

1.6.3 Document preamble

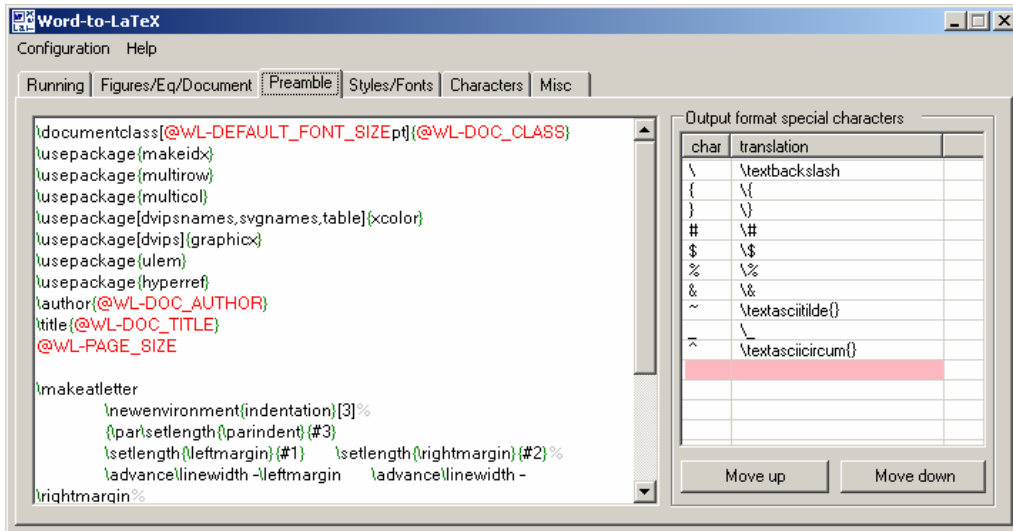


Figure 1.3: “Preamble” tab

Document preamble, inserted at the top of output files, can be easily edited in this dialog. Table 1.4 shows the list of macros that can be used in the preamble.

The translations of **Output format special characters** (e.g. “\” in L^AT_EX or “<” in XML) are defined in the right part of this dialog. Don’t forget to fill in these characters in the right order because some special characters can be used for the translation of other special characters (e.g. “\” must be at the top for L^AT_EX output). New characters can be added double-clicking the pink row.

1.6.4 Special characters

Special characters are divided into groups according to their Unicode [1] positions. Each character can have a **translation** used in regular text context and a **math translation** used in math context. Currently when a character has both translations defined, the text translation is always used. If it has only a math translation, the character is inserted as a simple inline equation. If no translation is defined, the character is inserted “as is” (in UTF-8 encoding).

The math translation does not influence the conversion of equations, which is completely defined in a TDL file (see section 1.6.2 for details).

Macro	Replaced with
@WL-DOC_CLASS	the Document class option from the previous dialog
@WL-DOC_AUTHOR	the input document's author (retrieved from the document's properties)
@WL-DOC_TITLE	the input document's title (retrieved from the document's properties)
@WL-PAGE_SIZE	see the Document settings in the previous section
@WL-DEFAULT_FONT_SIZE	the default font size; details in section 1.6.5
@WL-STYLE_COMMANDS	the commands created from paragraph and character user styles, see the Styles/Fonts tab in section 1.6.5 for details.

Table 1.4: Document preamble macros

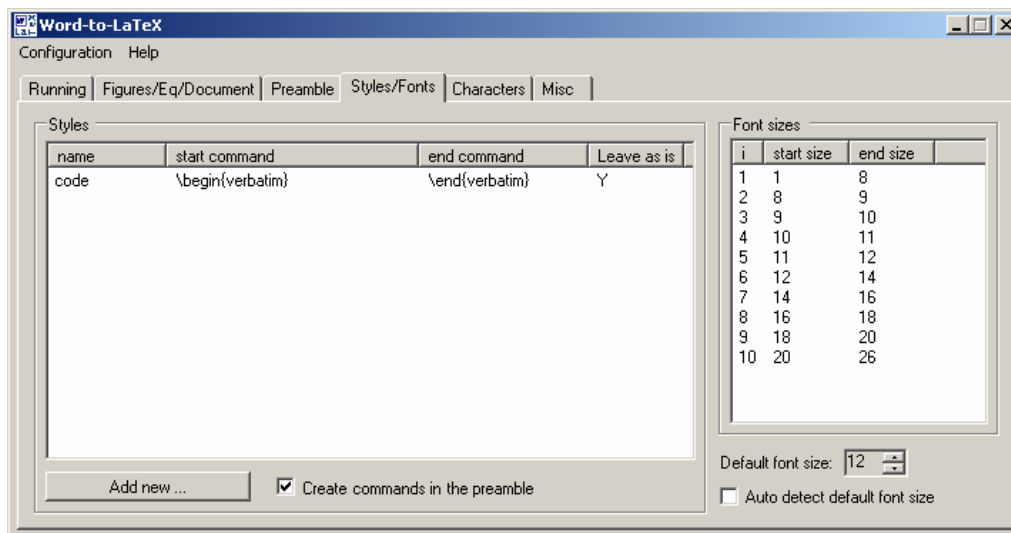


Figure 1.4: “Characters” tab

Default translations can be changed double-clicking the field you want to edit. The encoding of output files is UTF-8 which covers all national characters, so there is no need to define translations for Latin extended characters (e.g. “á”) or Cyrillic ones. Just make sure that you have appropriate commands in the document preamble, for example:

```
\usepackage[T2A]{fontenc}
\usepackage[utf8]{inputenc}
```

1.6.5 Styles and Font sizes

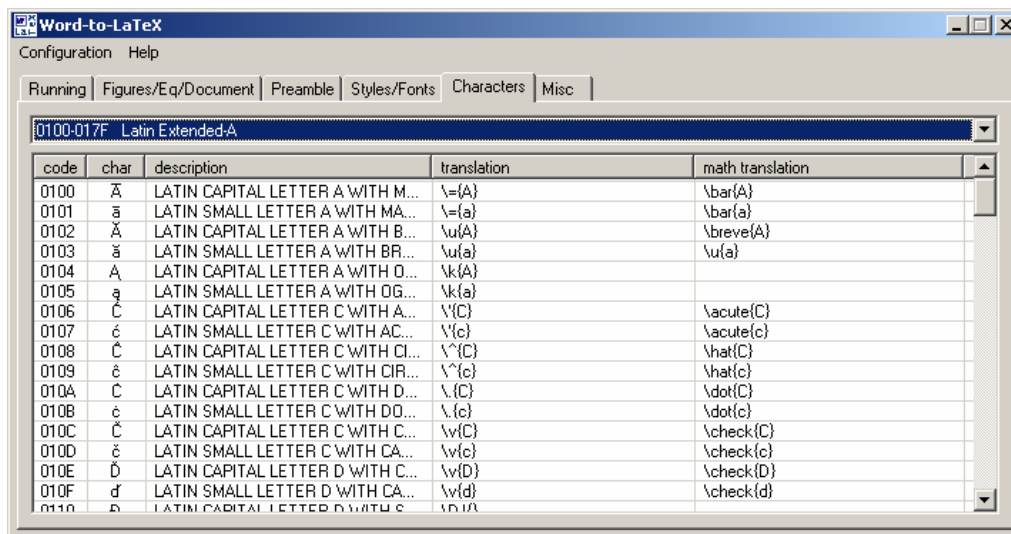


Figure 1.5: “Styles/Fonts” tab

The translations of paragraph and character user styles can be defined in this dialog. Press **Add new ...** and fill in the **name** of a style, the **start command** inserted before the text content of the style and the **end command** inserted after the text content. When you omit the definition of some style, appropriate commands will be created automatically on the basis of the style properties. *Word* built-in styles are skipped.

You can edit the list of styles double-clicking any of the fields. Write **Y** (or **N**) to the **leave as is** field if you don’t want to make any changes (character translations, wrapping) in the text content of the style. It’s suitable for styles that are translated to the *verbatim* environment.

Check **Create commands in the preamble** to make a special command for each style in the document preamble. It’s recommended to enable this option because it makes output files much more maintainable. For example, if you have a style named “code”, `\stylecode` command will be created and when you decide to change the definition of the style, you will do it only in one place.

Font sizes are split into 10 groups which are converted to the commands defined in **Translations** (see 1.6.2 for details). Each group has a point range of sizes that it covers – from the **start size** (exclusively) to the **end size** (inclusively). You can edit the default settings double-clicking the **end size** field of a group you want to change. Start sizes are counted automatically.

The portions of text that have the **Default font size** won't be marked with any command defining the font size. Therefore it's very important to have a correct value in this field to avoid a lot of unnecessary font size commands in the output file. Check **Auto detect default font size** to retrieve the default size from the *Word* built-in *Normal* style.

1.6.6 Miscellaneous options

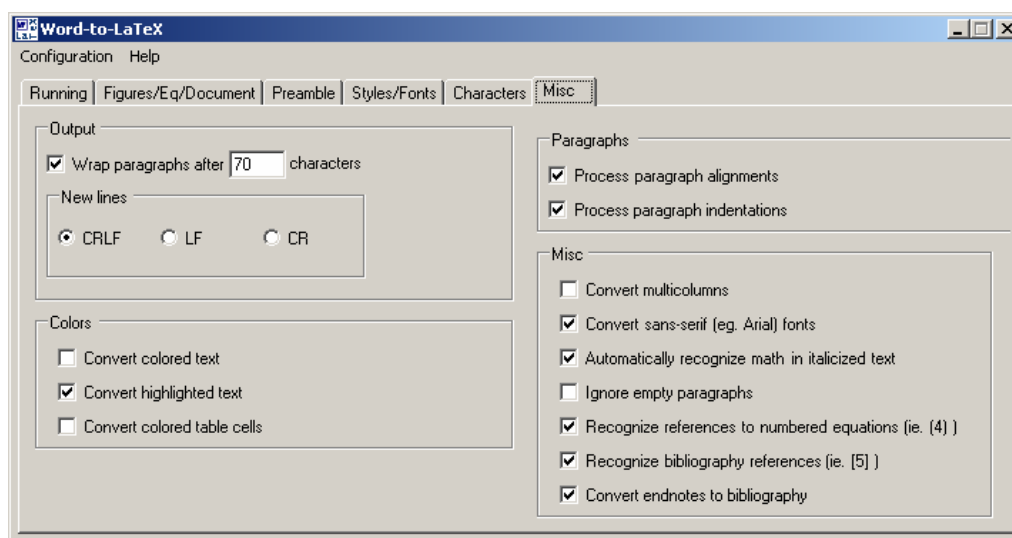


Figure 1.6: “Misc” tab

Output

Check **Wrap paragraphs** and insert an integer number to wrap the paragraphs in the output text file. The following line separators can be used in output files: **CRLF** (Windows), **LF** (Unix), **CR** (Macintosh).

Paragraphs

Check **Process paragraph alignments** and **Process paragraph indentations** to take them into account. Sometimes it's better to ignore *Word* alignments and indentations because \LaTeX can make them automatically and better.

Colors

Check **Convert colored text** to convert colored portions of text using `xcolor` package. But be very careful when checking this option because it takes a lot of time to find and convert the colored text.

The same package is used when you check **Convert highlighted text** (marked with the *Word* **Highlight** tool) and **Convert colored table cells**.

When any option is unchecked, it only means that commands defining colors won't be inserted into the output file. The whole text content will be, of course, converted.

Misc

Check **Convert multicolumns** to convert multicolumn sections inserted through **Format | Columns**. Sans-serif fonts like *Arial* or *Verdana* are converted to appropriate commands only when **Convert sans-serif fonts** is checked.

Check the option **Automatically recognize math in italicized text** and simple math expressions like i or $k < 30$ will be inserted as math text instead of text in italics.

The convertor can **Recognize references to numbered equations** if they match the pattern $([1-9]+)$ or $([1-9]+.[1-9]+)$ (e.g. (3.15)). A numbered equation must be inserted on a separate line and its label must be written at the right part of the same line. Any number of white space characters between the equation and its label is allowed.

Paragraphs not containing any text won't be converted when **Ignore empty paragraphs** is checked.

Word-to-L^AT_EX can **Convert endnotes into bibliography** items and **Recognize bibliography references** (citations) if they match the pattern $\[[A-Za-z0-9]+\]$ (e.g. [4] or [Ka76]). But if you don't use endnotes for bibliography items, you will still have to edit the bibliography section manually.

1.7 Running Word-to-L^AT_EX from Word

The conversion will be at least 10 times faster if you press the button on the *Word-to-L^AT_EX* toolbar installed directly into your *Word* application. The convertor interface is completely the same as the one described in the previous section.

If you have problems with running the convertor from *Word*, please verify that you have **Medium** or **Low** option checked in the *Word Tools* | **Macro** | **Security** menu.



Figure 1.7: *Word-to-L^AT_EX* toolbar in *Word*

1.8 Conversion to XML, XHTML, MathML

The output of the convertor completely depends on the configuration. There is no need to convert documents only to L^AT_EX. The `XMLConfig.xml` configuration file, stored in the *Word-to-L^AT_EX* directory, is used for conversion to XML [2] which is a nice intermediate format that can be easily transformed to whatever format you need. You should be familiar with XML and related technologies to understand a short overview.

The best way to insert mathematical equations into XML documents is MathML language. *Word-to-L^AT_EX* uses *MathType* built-in capability to export equations to MathML format.

XML format is very strict – XML files must be so-called “well-formed”. Sometimes the convertor produces a file that is not well-formed, but it's never difficult to correct such a file manually.

Once we have a well-formed XML file, an XSLT style [3] can be used to transform the file into the format we need. The `html.xsl` style, located in the *Word-to-L^AT_EX* directory, transforms the input file to XHTML format [4] combined with CSS [5]. This style was tested with *saxon* XSLT processor.

Appendix A

Sample documents

The following pages show two documents converted with *Word-to-L^AT_EX*.

1. Font styles

1.1. Styles 1

Lorem ~~ipsum dolor~~ sit amet, consectetur adipiscing elit. UT SED NISI vel justo lobortis venenatis. **Sed id risus. Donec sollicitudin.** Aenean nulla. Nam blandit, *sapien a venenatis viverra*, velit nisl mattis urna, non **luctus** sapien ante et leo. H₂O, E = mc²

1.2. Styles 2

Lorem *ipsum dolor* sit amet¹, consectetur adipiscing elit. Ut sed nisi vel justo lobortis venenatis. Sed id risus. Donec sollicitudin. Aenean nulla. **Nam blandit**, sapien a venenatis viverra, velit **nisl mattis** urna, **non luctus** sapien ante et leo.

2. Special characters in list

- Žluťoučký kůň úpěl ďábelské ódy.
 - Ψ Ω α ζ δ; i ∈ T; (a,b) ∉ A × B.

3. Paragraph indentation

 Lorem ipsum dolor sit amet,
 consectetur adipiscing elit.

 Lorem ipsum dolor sit amet, consectetur
 adipiscing elit. Ut sed nisi vel justo lobortis.

4. Simple table

Blue	Center bold	Right
2-1	<i>Italics</i>	Pink

5. Complex table

Header			
A	a	b	B
	c	d	

¹ Lorem ipsum dolor sit amet

1 Font styles

1.1 Styles 1

Lorem ~~ipsum dolor~~ sit amet, consectetur adipiscing elit. UT SED NISI vel justo lobortis venenatis. **Sed id risus**. Donec sollicitudin. Aenean nulla. Nam blandit, *sapient a venenatis viverra*, velit nisl mattis urna, non **luctus** sapien ante et leo. H₂O, E = mc²

1.2 Styles 2

Lorem *ipsum dolor* sit amet¹, consectetur adipiscing elit. Ut sed nisi vel justo lobortis venenatis. Sed id risus. Donec sollicitudin. Aenean nulla. **Nam blandit**, sapien a venenatis viverra, velit **nisl mattis** urna, **non luctus** sapien ante et leo.

2 Special characters in list

- Žluťoučký kůň úpěl ďábelské ódy.
 - $\Psi \Omega \alpha \zeta \delta; i \in T; (a,b) \notin A \times B$.

3 Paragraph indentation

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut sed nisi vel justo lobortis.

4 Simple table

Blue	Center bold	Right
2-1	<i>Italics</i>	Pink

5 Complex table

Header			
A	a	b	B
	c	d	

XML output transformed to HTML and rendered in *Mozilla*

Font styles

Styles 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. UT SED NISI vel justo lobortis venenatis. **Sed id risus. Donec sollicitudin.** Aenean nulla. Nam blandit, *sapientia venenatis viverra*, velit nisl mattis urna, non **luctus** sapien ante et leo. H₂O, E = mc²

Styles 2

Lorem *ipsum dolor* sit amet (Lorem ipsum dolor sit amet) , consectetur adipiscing elit. Ut sed nisi vel justo lobortis venenatis. Sed id risus. Donec sollicitudin. Aenean nulla. **Nam blandit**, sapien a venenatis viverra, velit **nisl mattis** urna, **non luctus** sapien ante et leo.

Special characters in list

- Žluťoučký kůň úpěl ďábelské ódy.
 - $\Psi \Omega \alpha \zeta \delta; i \in T; (a,b) \notin A \times B.$

Paragraph indentation

 Lorem ipsum dolor sit amet,
 consectetur adipiscing elit.

 Lorem ipsum dolor sit amet, consectetur
 adipiscing elit. Ut sed nisi vel justo lobortis.

Simple table

Blue	Center bold	Right
2-1	<i>Italics</i>	Pink

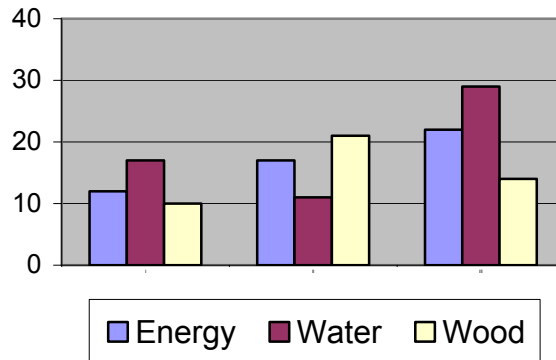
Complex table

Header			
A	a	b	B
	c	d	

Original *Word* document at the top, \LaTeX output compiled to PostScript at the bottom



Bitmap image



Microsoft Excel graph

Equation editor expressions

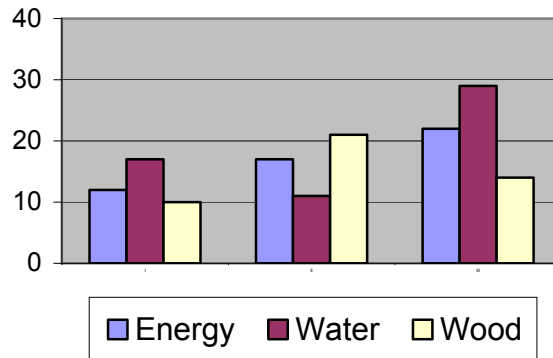
$$D(o_i, o_j) = \sum_{k=1}^{\max(l_i, l_j)} d(o_i^k, o_j^k) \quad (1)$$

Given a set of paths X_P and a set of path contents X_{PC} , **binary relation** $PPC \subseteq X_P \times X_{PC}$ is defined. An $\langle e, s \rangle \in PPC$ denotes the assignment of the path $e = e_1 / e_2 / \dots / e_k$ to the path content $s = s_1 / s_2 / \dots / s_k$.

3



Bitmap image



Microsoft Excel graph

Equation editor expressions

$$D(o_i, o_j) = \sum_{k=1}^{\max(l_i, l_j)} d(o_i^k, o_j^k) \quad (1)$$

Given a set of paths X_P and a set of path contents X_{PC} , **binary relation** $PPC \subseteq$

Appendix B

Structure of configuration files

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration xmlns='http://kebrt.cz/word-to-latex'
  xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'>
  <variousOptions>
    <option name="OUTPUT_FORMAT" value="latex" />
    <option name="EQUATIONS" value="toimages" />
    ...
  </variousOptions>
  <translationTable>
    <docElement name='FONT_BOLD'
      start='\textbf{' end='}' />
    <docElement name='HEADING1' start='\part{' end='}' />
    ...
  </translationTable>
  <specialChars>
    <latexChar char='\ ' convertTo='\textbackslash ' />
    ...
  </specialChars>
</configuration>
```

Figure B.1: Fragment of the `config.xml` configuration file

All the configuration is stored in an XML file with the `<configuration>` root element which contains three subelements:

<code><variousOptions></code>	various options applied during the conversion (output format, PostScript printer name, ...)
<code><translationTable></code>	table containing mappings between input document elements (sections, paragraphs, footnotes, and so on) and \LaTeX commands
<code><specialChars></code>	translation mappings between special (and national) characters and \LaTeX commands

B.1 Conversion options

All the options, listed in table B.1, belong to the `<variousOptions>` parent element. Each of the them is inserted into the `<option>` element with two attributes, `name` and `value`.

Option name	Description and possible values
ONLY_IMAGES	Convert only images and ignore text content. <ul style="list-style-type: none"> • <code>yes</code> × <code>no</code>
PRINTER_NAME	The name of a PostScript printer which is used for exporting images in EPS format. The printer driver has to be installed on your system. <ul style="list-style-type: none"> • e.g. <code>Generic Color PS</code>
IMAGE_FORMAT	The output format of images. <ul style="list-style-type: none"> • <code>eps</code> for EPS vector format; requires a PostScript printer • <code>png</code> for PNG bitmap format; not all the images can be exported as bitmaps
TDL_FILENAME	The translation file used for the conversion of equations. See the <code>Translators</code> subdirectory of your <i>MathType</i> directory for possible values (remember that <i>MathType</i> must be installed on your system to be able to convert equations). You can edit or add new files into this directory if you want to customize the conversion of equations. <ul style="list-style-type: none"> • e.g. <code>LaTeX.tdl</code>
EQUATIONS	The conversion of equations, covers <i>Equation Editor</i> , <i>MathType</i> and <code>EQ</code> fields equations. <ul style="list-style-type: none"> • <code>ignore</code> – do not convert • <code>convert</code> – convert using the translation file specified in the <code>TDL_FILENAME</code> option • <code>toimages</code> – convert to images
CREATE_COMMANDS_FOR_STYLES	The convertor will create (or not) new commands for paragraph and characters user styles in the preamble. Output text files are more maintainable if commands like <code>\code</code> are used instead of for example <code>\texttt</code> . <ul style="list-style-type: none"> • <code>yes</code> × <code>no</code>
DOC_CLASS	The <code>@WL-DOC_CLASS</code> macro used in the preamble will be replaced with the value of this option. <ul style="list-style-type: none"> • e.g. <code>article</code>

Table B.1: Conversion options

Option name	Description and possible values
OUTPUT_FORMAT	The format of output files. Please remember that all translations mappings described in B.2 should be set to match this output format. The convertor performs a few special actions depending on two possible values: <ul style="list-style-type: none"> • latex • xml
PAGE_SIZE	The <code>@WL-PAGE_SIZE</code> macro used in the document preamble will be replaced with the value of this option (only if the <code>PAGE_SIZE_PROCESSING</code> option is set to <code>my</code>). <ul style="list-style-type: none"> • e.g. <code>a4paper</code>
PAGE_SIZE_PROCESSING	Specifies how the page size will be processed, possible values are: <ul style="list-style-type: none"> • <code>complete</code> – the <code>@WL-PAGE_SIZE</code> macro used in the document preamble will be replaced with the complete page size definition matching the page size of the input document • <code>symbolic</code> – the convertor will try to translate the symbolic page size of the input document (e.g. <code>A4</code>) to an appropriate <code>L^AT_EX</code> size (e.g. <code>letterpaper</code>) • <code>my</code> – see the previous option
DEFAULT_FONT_SIZE	Defines the default font size of the input document. The portions of text having this size won't be marked with any font size command in the output file. Only integer numbers are allowed. <ul style="list-style-type: none"> • e.g. <code>12</code>
PARAGRAPH_ALIGNMENTS	Convert paragraph alignments. <p>– <code>yes</code> × <code>no</code></p>
PARAGRAPH_INDENTATION	Convert paragraph indentations. <p>– <code>yes</code> × <code>no</code></p>
COLOR_TEXT	Use special commands for colored text. <ul style="list-style-type: none"> • <code>yes</code> × <code>no</code>
COLOR_BG	Use special commands for text with colored background. <ul style="list-style-type: none"> • <code>yes</code> × <code>no</code>
COLOR_TABLE	Use special commands for table cells with colored background. <ul style="list-style-type: none"> • <code>yes</code> × <code>no</code>

Table B.1: Conversion options

Option name	Description and possible values
AUTO_DETECT_DEFAULT_FONT_SIZE	Detect the default font size of the input document automatically or not. The font size of the <i>Word</i> built-in <i>Normal</i> style will be taken as the default one if this option is set to yes . • yes × no
MULTICOLUMN	Convert multicolumn sections. • yes × no
WRAP_PARAGRAPHS	A positive value causes paragraphs to be wrapped into lines after each x characters. Any other value forces the convertor not to wrap paragraphs. • e.g. 80
NEW_LINE	Defines the line separator, possible values are: • crlf – Windows line separator • cr – Macintosh line separator • lf – Unix line separator
SANS_SERIF	Use special commands for sans-serif fonts. • yes × no
AUTO_RECOGNIZE_MATH	Recognize math expressions written in italics (e.g. <i>i</i>). • yes × no
IGNORE_EMPTY_PAR	Ignore paragraphs not containing any text. • yes × no
RECOGNIZE_NUMBERED_EQ_REF	Recognize references to numbered equations marked with labels like “(5)” or “(5.2)”. • yes × no
ENDNOTES_TO_BIBLIO	Convert endnotes to bibliography items. • yes × no
RECOGNIZE_BIBLIO_REF	Recognize in-text citations (references to bibliography items, e.g. “[4]”). – yes × no
FONT_SIZE[1-10]	These options define ranges for each converted font size group. The range for the i -th group is from $\text{FONT_SIZE}(i-1)+1$ to $\text{FONT_SIZE}(i)$ (inclusive). The first group (FONT_SIZE1) starts with the size 1. Only integer numbers are allowed. • e.g. 11 for the FONT_SIZE4 option and 12 for the FONT_SIZE5 option when the default font size is 12

Table B.1: Conversion options

B.2 Conversion mappings

Table B.2 shows the complete list of conversion mappings between input document elements (sections, paragraphs, lists, and so on) and *Word-to-L^AT_EX*. Each mapping has a start command (S:) which is inserted before the element and most of them have also an end command (E:) inserted after the element. Some elements like tabulators doesn't have any content, others hold some kind of content (text, equation, another element) which is inserted between the start and end command.

Names of macros that are specific to each element begin with “#”, macros common to all elements begin with “@”.

- @WL-NL new line
- @WL-TAB tabulator

Table B.2 also contains the default mappings for L^AT_EX and XML output. When E: is omitted, the end command is always ignored by the convertor, “—” stands for the empty translation command.

FONT_BOLD	bold font
S: \textbf{	
E: }	
S: 	
E: 	
FONT_ITALIC	italic font
S: \textit{	
E: }	
S: 	
E: 	
FONT_SMALLCAPS	small caps font
S: \textsc{	
E: }	
S: 	
E: 	
FONT_HIDDEN	hidden font
S: @WL-NL%	
E: @WL-NL	
S: 	
E: 	

Table B.2: Conversion mappings

FONT_SUBSCRIPT	subscript font
S: $\$_{\{$	
E: $\}$$	
S: <code></code>	
E: <code></code>	
FONT_SUPERSCRIPT	superscript font
S: $\$\^{$	
E: $\}$$	
S: <code></code>	
E: <code></code>	
FONT_COURIER	courier font (e.g. Courier, Courier New)
S: <code>\texttt{</code>	
E: <code>}</code>	
S: <code></code>	
E: <code></code>	
FONT_UPPERCASE	uppercase font
S: <code>\uppercase{</code>	
E: <code>}</code>	
S: <code></code>	
E: <code></code>	
FONT_UNDERLINE	underlined font
S: <code>\uline{</code>	
E: <code>}</code>	
S: <code></code>	
E: <code></code>	
FONT_DOUBLE_UNDERLINE	double-underlined font
S: <code>\uuline{</code>	
E: <code>}</code>	
S: <code></code>	
E: <code></code>	
FONT_WAVE_UNDERLINE	wavy-underlined font
S: <code>\uwave{</code>	
E: <code>}</code>	
S: <code></code>	
E: <code></code>	

Table B.2: Conversion mappings

FONT_STRIKE	strikethrough font
S: <code>\sout{</code>	
E: <code>}</code>	
S: <code></code>	
E: <code></code>	
FONT_SANS_SERIF	sans-serif font (e.g. Arial, Verdana)
S: <code>\textsf{</code>	
E: <code>}</code>	
S: <code></code>	
E: <code></code>	
FONT_SIZE1	font size (group 1)
S: <code>{\tiny</code>	
E: <code>}</code>	
S: <code><font-size value="1"></code>	
E: <code></font-size></code>	
FONT_SIZE2	font size (group 2)
S: <code>{\scriptsize</code>	
E: <code>}</code>	
S: <code><font-size value="2"></code>	
E: <code></font-size></code>	
FONT_SIZE3	font size (group 3)
S: <code>{\footnotesize</code>	
E: <code>}</code>	
S: <code><font-size value="3"></code>	
E: <code></font-size></code>	
FONT_SIZE4	font size (group 4)
S: <code>{\small</code>	
E: <code>}</code>	
S: <code><font-size value="4"></code>	
E: <code></font-size></code>	
FONT_SIZE5	font size (group 5)
S: <code>{\normalsize</code>	
E: <code>}</code>	
S: <code><font-size value="5"></code>	
E: <code></font-size></code>	

Table B.2: Conversion mappings

FONT_SIZE6	font size (group 6)
S: {\large	
E: }	
S: <font-size value="6">	
E: </font-size>	
FONT_SIZE7	font size (group 7)
S: {\Large	
E: }	
S: <font-size value="7">	
E: </font-size>	
FONT_SIZE8	font size (group 8)
S: {\LARGE	
E: }	
S: <font-size value="8">	
E: </font-size>	
FONT_SIZE9	font size (group 9)
S: {\huge	
E: }	
S: <font-size value="9">	
E: </font-size>	
FONT_SIZE10	font size (group 10)
S: {\Huge	
E: }	
S: <font-size value="10">	
E: </font-size>	
HEADING1	heading (level 1); headings have to be marked with the <i>Word</i> built-in styles; they can be defined up to level 9
S: \section{	
E: }	
S: <heading level="1">	
E: </heading>	
HEADING2	heading (level 2)
S: \subsection{	
E: }	
S: <heading level="2">	
E: </heading>	

Table B.2: Conversion mappings

HEADING3	heading (level 3)
S: <code>\subsubsection{</code>	
E: <code>}</code>	
S: <code><heading level="3"></code>	
E: <code></heading></code>	
ALIGN_CENTER	paragraph alignment – centered
S: <code>\begin{center}@WL-NL</code>	
E: <code>@WL-NL\end{center}</code>	
S: <code><align type="center" /></code>	
E: —	
ALIGN_LEFT	paragraph alignment – left
S: <code>{\raggedright@WL-NL</code>	
E: <code>@WL-NL}</code>	
S: <code><align type="left" /></code>	
E: —	
ALIGN_RIGHT	paragraph alignment – right
S: <code>{\raggedleft@WL-NL</code>	
E: <code>@WL-NL}</code>	
S: <code><align type="right" /></code>	
E: —	
TABLE_ALIGN_CENTER	table paragraph alignment – centered
• #WIDTH	table cell width (in points)
S: <code>\parbox{#WIDTHpt}{\centering</code>	
E: <code>}</code>	
S: <code><align type="center" /></code>	
E: —	
TABLE_ALIGN_LEFT	table paragraph alignment – left
• #WIDTH	table cell width (in points)
S: <code>\parbox{#WIDTHpt}{\raggedright</code>	
E: <code>}</code>	
S: <code><align type="left" /></code>	
E: —	

Table B.2: Conversion mappings

TABLE_ALIGN_RIGHT	table paragraph alignment – right
• #WIDTH	table cell width (in points)
S: <code>\parbox{#WIDTHpt}{\raggedleft</code>	
E: <code>}</code>	
S: <code><align type="right" /></code>	
E: <code>—</code>	
FOOTNOTE	footnote
S: <code>\footnote{</code>	
E: <code>}</code>	
S: <code><footnote></code>	
E: <code></footnote></code>	
PAGE_BREAK	page break
S: <code>\pagebreak{ }@WL-NL@WL-NL</code>	
S: <code><pagebreak /></code>	
EQUATION_INLINE	inline equation
S: <code>\begin{math}</code>	
E: <code>\end{math}</code>	
S: <code><equation type="inline"></code>	
E: <code></equation></code>	
EQUATION_NUMBERED	numbered equation
• #ORIG_LABEL	original equation label retrieved from the input document
S: <code>\begin{equation}</code>	
E: <code>@WL-NL%#ORIG_LABEL@WL-NL\end{equation}</code>	
S: <code><equation type="numbered" origlabel="#ORIG_LABEL"></code>	
E: <code></equation></code>	
EQUATION_LABEL	equation label inserted into the EQUATION_NUMEBERED element
• #NAME	auto-generated label (auto-incrementing counter is used)
S: <code>\label{#NAME}</code>	
S: <code><label name="#NAME"/></code>	

Table B.2: Conversion mappings

EQUATION_OUTLINE	equation displayed on a separate line
S: <code>\begin{displaymath}</code>	
E: <code>\end{displaymath}</code>	
S: <code><equation type="outline"></code>	
E: <code></equation></code>	
INDEX_ENTRY	index entry (<i>Word</i> XE field)
S: <code>\index{</code>	
E: <code>}</code>	
S: <code><index-entry></code>	
E: <code></index-entry></code>	
INDEX	index (<i>Word</i> INDEX field), \LaTeX generates the whole index automatically
S: <code>\printindex</code>	
S: <code><printindex /></code>	
IMAGE_COMMAND	image
• #WIDTH	image width (in points)
• #FILENAME	auto-generated image filename (e.g. <code>img1.eps</code>)
• #TITLE	image title (if present)
S: <code>\includegraphics[width=#WIDTHpt]{#FILENAME}@WL-NL</code>	
S: <code><image width="#WIDTH" src="#FILENAME" title="#TITLE" /></code>	
IMAGE_CONTAINER	image container (used when the image has a title)
S: <code>\begin{figure}[h]@WL-NL</code>	
E: <code>\end{figure}</code>	
S: <code>—</code>	
E: <code>—</code>	
IMAGE_TITLE	image title inserted into the <code>IMAGE_CONTAINER</code> element
• #TITLE	title
S: <code>\caption{#TITLE}</code>	
S: <code>—</code>	
TOC	table of contents (<i>Word</i> TOC field), \LaTeX generates the table of contents automatically as well as <i>Word</i>
S: <code>\tableofcontents</code>	
S: <code><table-of-contents /></code>	

Table B.2: Conversion mappings

HYPERLINK	hyperlink
<ul style="list-style-type: none"> • #HREF 	hyperlink target; the macro can be used also in the end command
S: <code>\href{#HREF}{</code>	
E: <code>}</code>	
S: <code><link href="#HREF"></code>	
E: <code></link></code>	
SPECIAL_COMMAND	L ^A T _E X command(s) inserted into the document through the <i>Word</i> PRIVATE field whose content must begin with the case-insensitive string <code>latex:</code> , such a field may look like this: PRIVATE LaTeX: <code>\indent</code> (<code>\indent</code> will be inserted between the start and end command)
S: <code>—</code>	
E: <code>—</code>	
S: <code>—</code>	
E: <code>—</code>	
REFERENCE	bookmark reference
<ul style="list-style-type: none"> • #NAME 	name of the bookmark that is being referenced
S: <code>\ref{#NAME}</code>	
S: <code><reference name="#NAME" /></code>	
MATH_REFERENCE	equation reference; the <i>Word</i> hard-coded reference (e.g. “(3)”) will be the content of this element
<ul style="list-style-type: none"> • #NAME 	name of the equation that is being referenced, it is generated for each numbered equation in the document (e.g. “eq3”).
S: <code>(\ref{#NAME})@WL-NL%</code>	
E: <code>@WL-NL</code>	
S: <code><math-reference name="#NAME"></code>	
E: <code></math-reference></code>	
NOTE_REFERENCE	note reference; currently only endnotes are supported
<ul style="list-style-type: none"> • #NAME 	name of the note (typically number) that is being referenced
S: <code>\cite{ref#NAME}</code>	
S: <code><note-reference name="#NAME" /></code>	

Table B.2: Conversion mappings

BIBLIO_REFERENCE	reference to a bibliography item (“citation”); the <i>Word</i> hard-coded citation (e.g. “[Ka75]”) will be the content of this element
• #NAME	name of the bibitem (e.g. “Ka75”)
S: \cite{ref#NAME}@WL-NL%	
E: @WL-NL	
S: <biblio-reference name="#NAME">	
E: </biblio-reference>	
PAGE_REFERENCE	page reference
• #NAME	name of the bookmark that is being referenced
S: \pageref{#NAME}	
BOOKMARK_LABEL	bookmark
• #NAME	name of the bookmark
S: \label{#NAME}	
S: <bookmark name="#NAME" />	
STYLE	paragraph or character user style
• #NAME	name of the style; all numbers in the name are replaced with words (e.g. “1” → “One”)
S: \#NAME{	
E: }	
S: <style name="#NAME">	
E: </style>	
STYLE_DEFINITION	container for a single user style definition; commands describing the style will be inserted into
• #NAME	name of the user style
S: \newcommand{\#NAME}[1]{	
E: }	
S: <style-definition name="#NAME">	
E: </style-definition>	
DOCUMENT_BODY	document body
S: \begin{document}@WL-NL	
E: \end{document}	
S: <body>	
E: </body></document>	

Table B.2: Conversion mappings

LIST_ENUMERATE	enumerated list
S: <code>\begin{enumerate}@WL-NL</code>	
E: <code>\end{enumerate}@WL-NL@WL-NL</code>	
S: <code>@WL-NL<list type="enumerate"></code>	
E: <code></list>@WL-NL</code>	
LIST_ITEMIZE	itemized list
S: <code>\begin{itemize}@WL-NL</code>	
E: <code>\end{itemize}@WL-NL@WL-NL</code>	
S: <code>@WL-NL<list type="itemize"></code>	
E: <code></list>@WL-NL</code>	
LIST_ITEM	list item
S: <code>@WL-TAB\item</code>	
E: <code>—</code>	
S: <code><list-item></code>	
E: <code></list-item>@WL-NL</code>	
PARAGRAPH	common paragraph
S: <code>—</code>	
E: <code>@WL-NL@WL-NL</code>	
S: <code>@WL-NL<para></code>	
E: <code></para>@WL-NL</code>	
TABLE_PARAGRAPH	paragraph in a table
S: <code>@WL-NL</code>	
E: <code>@WL-NL</code>	
S: <code>@WL-NL<table-para></code>	
E: <code></table-para>@WL-NL</code>	
LIST_PARAGRAPH	paragraph in a list
S: <code>—</code>	
E: <code>@WL-NL</code>	
S: <code><list-para></code>	
E: <code></list-para></code>	
LINE_BREAK	line break
S: <code>@WL-NL\\@WL-NL</code>	
S: <code><linebreak /></code>	
TAB	tabulator
S: <code>\hspace{15pt}</code>	
S: <code><tab /></code>	

Table B.2: Conversion mappings

TABLE_CELL	table cell
• #WIDTH	cell width
S: &	
E: —	
S: <table-cell width="#WIDTH">	
E: </table-cell>	
TABLE_ROW	table row
S: —	
E: \\@WL-NL	
S: <table-row>	
E: </table-row>	
TABLE	table
• #TITLE	title of the table
S: @WL-NL\vspace{3pt} \noindent@WL-NL\begin{tabular}	
E: \end{tabular}\\@WL-NL\vspace{2pt}@WL-NL	
S: @WL-NL<table title="#TITLE">	
E: </table>@WL-NL	
TABLE_CONTAINER	table container (used when the table has a title)
S: @WL-NL\begin{table}[h]	
E: \end{table}@WL-NL	
S: —	
E: —	
TABLE_TITLE	table title inserted into the TABLE_CONTAINER element
• #TITLE	title
S: \caption{#TITLE}	
S: —	
TABLE_MULTIRROW	table cell with merged rows
• #ROWS	number of merged rows in the cell
S: \multirow{#ROWS}{*}{	
E: }	
S: <table-multirow-cell multi="#ROWS" />	
E: —	

Table B.2: Conversion mappings

TABLE_CELL_COLOR	command for the colored background of table cells; the #COLOR macro in the next element (TABLE_MULTI_COLUMN) will be replaced with this command
<ul style="list-style-type: none"> • #COLOR 	background color in HTML notation (e.g. FF0000)
S: >{\columncolor[HTML]{#COLOR}}	
S: color="#COLOR"	
TABLE_MULTICOLUMN	table cell with merged columns
<ul style="list-style-type: none"> • #COLS • #LEFT_BORDER • #RIGHT_BORDER • #COLOR • #ALIGN 	<ul style="list-style-type: none"> number of merged columns “ ” if the cell has a left border “ ” if the cell has a right border see the previous element cell content alignment; l (left), r (right), c (center)
S: \multicolumn{#COLS}{#LEFT_BORDER#COLOR#ALIGN#RIGHT_BORDER}{	
E: }	
S: <table-cell multi="#COLS" left-border="#LEFT_BORDER" right-border="#RIGHT_BORDER" align="#ALIGN" width="#WIDTH" #COLOR>	
E: </table-cell>	
PAR_INDENT	paragraph indentation
<ul style="list-style-type: none"> • #LEFT_INDENT • #RIGHT_INDENT • #FIRST_LINE_INDENT 	<ul style="list-style-type: none"> left indentation (in points) right indentation (in points) first line indentation (in points)
S: \begin{indentation}{#LEFT_INDENTpt}{#RIGHT_INDENTpt}{#FIRST_LINE_INDENTpt}@WL-NL	
E: @WL-NL\end{indentation}	
S: @WL-NL<par-indent left="#LEFT_INDENT" right="#RIGHT_INDENT" first-line="#FIRST_LINE_INDENT" />@WL-NL	
E: —	
MULTICOLUMN	multicolumn section
<ul style="list-style-type: none"> • #COLS 	number of columns in the section
S: \begin{multicols}{#COLS}	
E: \end{multicols}	
S: <multicol count="#COLS">	
E: </multicol>	

Table B.2: Conversion mappings

COLOR_TEXT	colored text
• #COLOR	color in HTML notation (e.g. FF0000)
S: \textcolor[HTML]{#COLOR}{	
E: }	
S: <font-color color="#COLOR">	
E: </font-color>	
COLOR_BG	text with colored background
• #COLOR	color in HTML notation (e.g. FF0000)
S: \colorbox[HTML]{#COLOR}{	
E: }	
S: <font-background color="#COLOR">	
E: </font-background>	
ENDNOTES_SECTION	container for endnotes, can be used for inserting the bibliography
S: \begin{thebibliography}{99}@WL-NL	
E: \end{thebibliography}@WL-NL	
S: <bibliography>	
E: </bibliography>	
ENDNOTE	endnote, this translation is used in the ENDNOTES_SECTION context, suitable for inserting a single bibliography item
• #NUMBER	number of the endnote
S: @WL-TAB\bibitem[#NUMBER]{ref#NUMBER}	
E: @WL-NL	
S: @WL-TAB<bib-item name="#NUMBER">	
E: </bib-item>	
ENDNOTE_REFERENCE	endnote, this translation is used at the endnote's insertion point
• #NUMBER	number of the endnote
• #CONTENT	endnote's text content (can be used when translating endnotes to footnotes)
S: \cite{ref#NAME}	
S: <endnote-reference name="#NUMBER" />	

Table B.2: Conversion mappings

COLOR_BG_AND_BORDER	text with colored border and background
<ul style="list-style-type: none"> • #BORDER_COLOR 	border color, in HTML notation (e.g. FF0000)
<ul style="list-style-type: none"> • #COLOR 	text color, dtto
S: \fcolorbox[HTML]{#BORDER_COLOR}[HTML]{#COLOR}{	
E: }	
S: <box border-color="#BORDER_COLOR" background-color="#COLOR">	
E: </box>	
COLOR_BORDER	colored border around text
<ul style="list-style-type: none"> • #BORDER_COLOR 	border color, in HTML notation (e.g. FF0000)
S: \fcolorbox[HTML]{#BORDER_COLOR}[HTML]{FFFFFF}{	
E: }	
S: <box border-color="#BORDER_COLOR">	
E: </box>	
BORDER	black border around text
S: \fbox{	
E: }	
S: <box>	
E: </box>	

Table B.2: Conversion mappings

B.3 Special characters

The configuration of special characters is enclosed in the `<specialChars>` element. `<latexChar>` elements are used for defining characters that have a special meaning in the output format. They must be written in a correct order because one special character can be used for translating another special character which is illustrated in the following example.

```
<latexChar char='\ ' convertTo='\textbackslash ' />
<latexChar char='{ ' convertTo='\{' />
```

All the other special and national characters are defined in `<char>` elements. The `code` attribute contains the Unicode [1] number of each character. The details about the common context translation (`convertTo` attribute) and the math context translation (`mathConvertTo` attribute) can be found in section 1.6.4. A short example follows.

```
<char code="010C" convertTo="\v{C}" mathConvertTo="\check{C}" />
<char code="010D" convertTo="\v{c}" mathConvertTo="\check{c}" />
```

Bibliography

- [1] *Unicode Home Page*, <http://www.unicode.org/>
- [2] *Extensible Markup Language (XML)*, <http://www.w3.org/XML/>
- [3] *XSL Transformations (XSLT)*, <http://www.w3.org/TR/xslt>
- [4] *XHTML 1.0 The Extensible HyperText Markup Language*,
<http://www.w3.org/TR/xhtml1/>
- [5] *Cascading Style Sheets*, <http://www.w3.org/Style/CSS/>