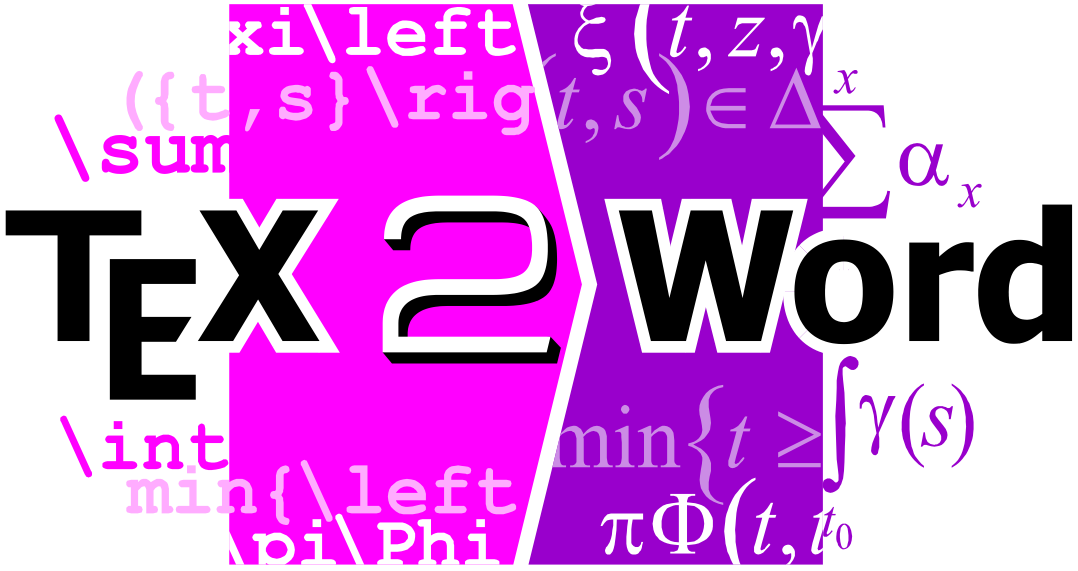
Chikrii Softlab
www.chikrii.com



User Manual

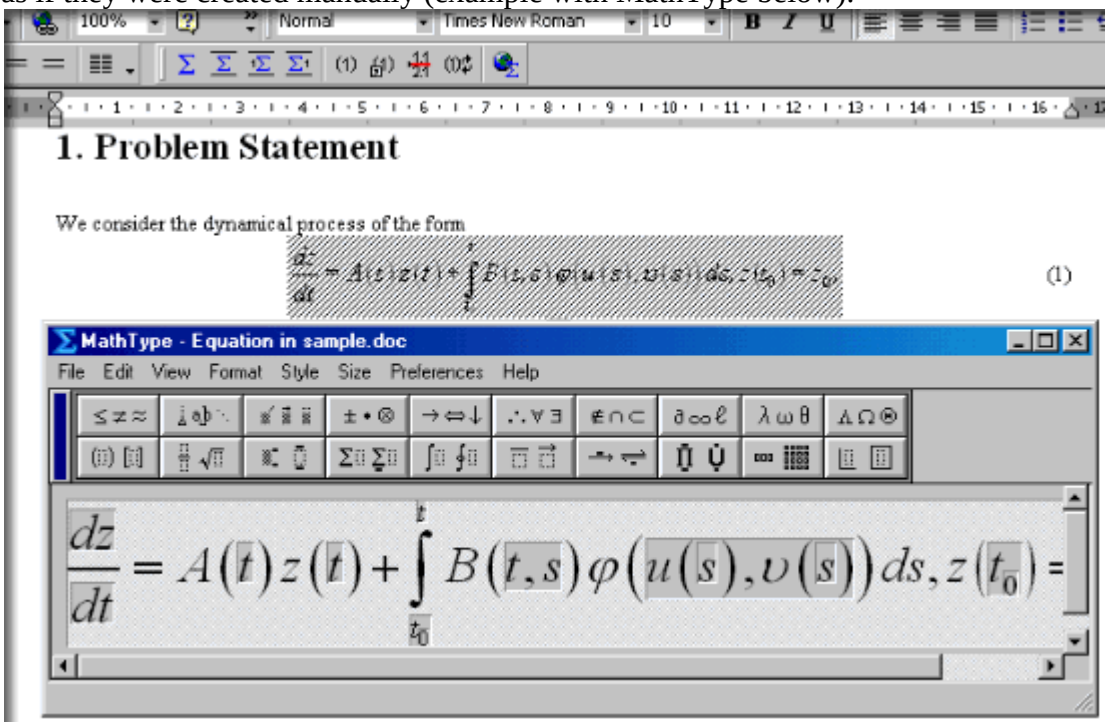
Contents

1 Introduction

1.1 About TeX2Word

TeX2Word is a converter designed in order to use with Microsoft Word and enables Microsoft Word to open documents in TeX format (this includes any TeX-based formats like LaTeX, etc.) This gives the opportunity to convert existing TeX/LaTeX documents to Microsoft Word.

TeX2Word translates TeX/LaTeX mathematical expressions to fully-functional equations (MathType or Office 2007/2010 built-in Equation Editor), so that you can double-click and edit them as if they were created manually (example with MathType below):



1.2 What For?

Why you may need your documents to be in Microsoft Word format? TeX/LaTeX is a de-facto standard in scientific publishing, but many publishers accept papers only in Microsoft Word format. Other reason is wide spectrum of software compatible with Microsoft Word, for example, Microsoft Word documents can be published on the Web, or saved as XML.

2 Getting Started

2.1 Overview

This part describes the steps you need to go through to get TeX2Word up and running.
TeX2Word system requirements and installation instructions are listed below. Although installing TeX2Word is simply a matter of running its Setup program and following a few simple instructions, you may want to read this part first so you have a better understanding of TeX2Word and its components.

2.2 System Requirements

In order to install and run TeX2Word, your computer must have:

- ✓ Microsoft Windows XP SP2 or later (Vista, 7,…);
- ✓ Any 32-bit Microsoft Word/Office;
- ✓ If your Microsoft Word/Office is older than 2007 version - Design Science MathType must be installed (version 4 or later);
- ✓ A hard disk drive with at least 100 megabytes of free space.

2.3 Installing TeX2Word

Installing TeX2Word is very simple – just run TeX2Word Setup program (file: tex2word.exe) and follow its instructions.

2.3.1 TeX2Word Setup

Once you have started TeX2Word Setup, just follow the instructions presented to you. Following components will be installed:

- The TeX2Word converter for Microsoft Word (file: tex2word.cnv);
- TeX2Word User Manual (file: tex2word.pdf);
- TeX2Word License Agreement (file: license.txt);
- Windows "Add/Remove Programs" uninstallation support (file: uninstall.exe);
- pseudoTeX macro files which build all TeX2Word knowledge about TeX/LaTeX formats and packages (files: *.ptex in pseudoTeX subfolder).

You can find all these files in TeX2Word Destination folder when TeX2Word is installed.

If "TeX" option doesn't appear in Word's "File | Open…" dialog, it might be because Microsoft Office/Word Text Converter folder is marked read-only. You may copy the file tex2word.cnv manually to Text converter folder (usually Program Files\Common Files\Microsoft Shared\TEXTCONV). After doing this you'll have to open any text file (it is important that this should be non-native Word doc) in Word, restart Word, and "TeX" option will appear.

2.3.2 Uninstalling TeX2Word

TeX2Word Setup supports Windows "Add/Remove Programs" feature:

- Click the "Start" button, point to "Settings", and then click "Control Panel";
- Double-click on "Add/Remove Programs";
- Follow the instructions on your screen.

2.4 Upgrading from previous version

There's no need to uninstall TeX2Word when upgrading to newer version. Just install TeX2Word as usual and all updates will be made automatically.

3 Basic Usage or How do I use it?

Once TeX2Word installed, its operation is seamless, below is shown a three-step procedure of importing TeX/LaTeX document to Microsoft Word:

1. Start Microsoft Word (if it's not already running);
2. Invoke "File | Open…" dialog box and choose "TeX" format:

3. Choose file you want to be imported and click Open

That's all!

Following condition must take place: all files (pictures, subdocuments, etc.) must be located in the same folder as TeX/LaTeX document you are going to import.

4 Advanced Usage

4.1 Overview

TeX2Word includes built-in pseudoTeX compiler which processes your TeX/LaTeX documents. pseudoTeX compiler doesn't use LaTeX packages or class/style files possibly presented in your system. Instead it has own definition files (with filename extension .ptex) located in pseudoTeX subfolder of TeX2Word program folder. These files are written in pseudoTeX language which itself represents some sort of superset of TeX language.

pseudoTeX always starts processing from file pseudotex.ptex and till command **\dothejob**, which loads your document into pseudoTeX input. After loading some "pseudo" package or defining commands which cover some LaTeX package pseudoTeX macro files always use command

\api@lockfile <filename>

which protects pseudoTeX from duplicate processing.

The main aim of TeX2Word is to translate structural elements of your TeX/LaTeX document to appropriate Word structures/elements. And that's why pseudoTeX introduces about hundred or so new "primitives".

4.2 Stylesheet primitives

pseudoTeX primitive	Parameter syntax	Meaning
\stylesheet@colortbl	**<N>[=] rgb <R> <G> <B> or <N>[=] cmyk <C> <M> <Y> <K>**	**defines entry number N in color table with value defined in RGB or CMYK color model**
\stylesheet@colortbl@clear	**N/A**	**clears previous color table definitions**
\stylesheet@fonttbl	**<N>[=] [FAMILY] <font name>;**	**defines entry number N in font table; font family specification FAMILY is optional and can be one of the following: roman, swiss, modern, script, décor, tech, bidi**
\stylesheet@fonttbl@clear	**N/A**	**clears previous font table definitions**
\stylesheet@style	**<N>[=]<name>;**	**defines paragraph style with reference number N in paragraph style table**
\stylesheet@clear	**N/A**	**clears previous paragraph style definitions**

4.3 Top-level markup primitives

pseudoTeX primitive	Parameter syntax	Meaning
\entity@toplevel@element@markup@requiredpagebreak	**N/A**	**inserts mandatory page-break**
\entity@toplevel@element@markup@requiredcolumnbreak	**N/A**	**inserts mandatory column-break**
\entity@toplevel@element@markup@nonrequiredpagebreak	**N/A**	**inserts optional page-break**
\entity@toplevel@element@markup@nonrequiredcolumnbreak	**N/A**	**inserts optional column-break**

4.4 Paragraph properties primitives

pseudoTeX primitive	Parameter syntax	Meaning
\entity@paragraph@just	**[=]<N>**	**sets current paragraph justification: 0 – justified, 1 – left, 2 – right, 3 – centered**
\entity@paragraph@left	**[=]<N>**	**left-indent**
\entity@paragraph@right	**[=]<N>**	**right-indent**
\entity@paragraph@first	**[=]<N>**	**first line indent**
\entity@paragraph@style	**[=]<N>**	**sets current paragraph style; N is reference to paragraph style table**
\entity@paragraph@keepintact	**[=]<N>**	**self-explaining**
\entity@paragraph@keepwithnext	**[=]<N>**	**self-explaining**
\entity@paragraph@level	**[=]<N>**	**self-explaining**
\entity@paragraph@outlinelevel	**[=]<N>**	**self-explaining**
\entity@paragraph@pagebreakbefore	**[=]<N>**	**self-explaining**
\entity@paragraph@sidebyside	**[=]<N>**	**self-explaining**
\entity@paragraph@noline	**[=]<N>**	**self-explaining**
\entity@paragraph@spacebefore	**[=]<N>**	**self-explaining**
\entity@paragraph@spaceafter	**[=]<N>**	**self-explaining**
\entity@paragraph@spacebetween	**[=]<N>**	**self-explaining**
\entity@paragraph@linespacingmultiple	**[=]<N>**	**self-explaining**
\entity@paragraph@righttoleftreading	**[=]<N>**	**self-explaining**
\entity@paragraph@sbasedon	**[=]<N>**	**reference to style on which this paragraph style is based**
\entity@paragraph@snext	**[=]<N>**	**next style after this**

4.5 Paragraph-level markup primitives

pseudoTeX primitive	Parameter syntax	Meaning
\entity@paragraph@element@linebreak	**N/A**	**inserts line-break**
\entity@paragraph@element@nonbreakingspace	**N/A**	**inserts non-breaking space**
\entity@paragraph@element@enspace	**N/A**	**En-space**
\entity@paragraph@element@tab	**N/A**	**tab-mark**
\entity@paragraph@element@emdash	**N/A**	**---**
\entity@paragraph@element@endash	**N/A**	**--**
\entity@paragraph@element@emspace	**N/A**	**Em-space**
\entity@paragraph@element@bullet	**N/A**	**•**
\entity@paragraph@element@lquote	**N/A**	**’**
\entity@paragraph@element@rquote	**N/A**	**‘**
\entity@paragraph@element@ldblquote	**N/A**	**“**

pseudoTeX primitive	Parameter syntax	Meaning
\entity@paragraph@element@rdblquote	**N/A**	**”**
\entity@paragraph@element@optionalhyphen	**N/A**	**self-explaining**
\entity@paragraph@element@nonbreakinghyphen	**N/A**	**self-explaining**
\entity@paragraph@element@zerowidthjoiner	**N/A**	**self-explaining**
\entity@paragraph@element@zerowidthnonjoiner	**N/A**	**self-explaining**
\entity@paragraph@new	**N/A**	**starts new paragraph**
\entity@paragraph@element@picture	**[=]<filename>;**	**inserts picture**
\entity@toc@placeholder	**N/A**	**inserts placeholder for table of contents**
\entity@paragraph@footnote@begin	**N/A**	**starts footnote**
\entity@paragraph@footnote@end	**N/A**	**ends footnote**
\entity@paragraph@endnote@begin	**N/A**	**starts endnote**
\entity@paragraph@endnote@end	**N/A**	**ends endnote**
\entity@paragraph@hyperlink@begin	**N/A**	**starts hyperlink**
\entity@paragraph@hyperlink@text	**N/A**	**separates hyperlink title**
\entity@paragraph@hyperlink@end	**N/A**	**ends hyperlink**
\entity@index@entry@begin	**N/A**	**starts index entry**
\entity@index@entry@end	**N/A**	**ends index entry**
\entity@index@entry@subentry	**N/A**	**starts index sub-entry**
\entity@index@placeholder	**N/A**	**insers placeholder for index**

4.6 Paragraph span primitives

pseudoTeX primitive	Parameter syntax	Meaning
\entity@paragraph@span@bold	**[=]<N>**	**0 – turns off “bold”, 1 – turns on “bold” font attribute**
\entity@paragraph@span@italic	**[=]<N>**	**0 – turns off “*italic*”, 1 – turns on “*italic*” font attribute**
\entity@paragraph@span@underline	**[=]<N>**	**0 – turns off all underlining, 1 - continuous underline, 2 - dotted underline, 3 - dash underline, 4 - dot dash underline, 5 - dot dot dash underline, 6 - double underline, 7 - thick underline, 8 - word underline, 9 - wave underline**
\entity@paragraph@span@font	**[=]<N>**	**sets current font to font from entry N in font table**

pseudoTeX primitive	Parameter syntax	Meaning
\entity@paragraph@span@allcaps	**[=]<N>**	**0 – turns off "all capitals" font attribute, 1 – turns on "all capitals" font attribute**
\entity@paragraph@span@fontsize	**[=]<N>**	**sets current font size to N half-points**
\entity@paragraph@span@strike	**[=]<N>**	**sets "strikethrough" font attribute to one of the following: 0 – none, 1 – strikethrough, 2 - double strikethrough**
\entity@paragraph@span@cf	**[=]<N>**	**defines current foreground color, N is reference to color table entry**
\entity@paragraph@span@cb	**[=]<N>**	**defines current background color, N is reference to color table entry**
\entity@paragraph@span@outline	**[=]<N>**	**0 – turns off "outline" font attribute, 1 – turns it on**
\entity@paragraph@span@smallcaps	**[=]<N>**	**0 – turns off "small capitals" font attribute, 1 – turns it on**
\entity@paragraph@span@shadow	**[=]<N>**	**0 – tuns off "shadow" font attribute, 1 – turns it on**
\entity@paragraph@span@ignoreligatures	**[=]<N>**	**0 – turns off TeX ligature filter, 1 – turns it on**
\entity@paragraph@span@animtext	**[=]<N>**	**sets text animation attribute as following: 0 – none, 1 – Las Vegas Lights, 2 – Blinking background, 3 – Sparkle text, 4 – Marching black ants, 5 – Marching red ants, 6 – Shimmer**
\entity@paragraph@span@emboss	**[=]<N>**	**0 – none, 1 – Emboss**
\entity@paragraph@span@engrave	**[=]<N>**	**0 – none, 1 – Engrave**
\entity@paragraph@span@script	**[=]<N>**	**0 – turns off superscripting or subscripting, 1 – subscripts text, 2 – superscripts text**
\entity@paragraph@span@hidden	**[=]<N>**	**1 – marks text as hidden,**

pseudoTeX primitive	Parameter syntax	Meaning
\entity@paragraph@span@lang	**[=]<N>**	**0 – marks text as visible specifies current content language (N is RTF language code)**
\entity@paragraph@span@casing	**[=]<N>**	**0 – none, 1 – uppercase, 2 – lowercase text**
\entity@paragraph@span@codepage	**[=]<N>**	**specifies current content Windows codepage**

4.7 List primitives

pseudoTeX primitive	Parameter syntax	Meaning
\entity@list@begin	**N/A**	**starts new LIST**
\entity@list@end	**N/A**	**ends LIST**
\entity@list@type	**[=]<N>**	**defines type of current LIST: 0 (default) – numbered list, 1 – bulleted list**
\entity@list@item@begin	**N/A**	**starts new LIST ITEM**
\entity@list@item@end	**N/A**	**ends LIST ITEM**

4.8 Table primitives

pseudoTeX primitive	Parameter syntax	Meaning
\entity@table@begin	**N/A**	**starts new TABLE (or nested TABLE)**
\entity@table@end	**N/A**	**ends TABLE (or nested TABLE)**
\entity@table@row@begin	**N/A**	**starts TABLE ROW**
\entity@table@row@end	**N/A**	**ends TABLE ROW**
\entity@table@row@omit	**N/A**	**kills last defined TABLE ROW**
\entity@table@cell@begin	**N/A**	**starts TABLE CELL**
\entity@table@cell@end	**N/A**	**ends TABLE CELL**
\entity@table@cell@colspan	**[=]<N>**	**spans current CELL to N columns**
\entity@table@cell@rowspan	**[=]<N>**	**spans current CELL to N rows**

4.9 Cross-referencing primitives

pseudoTeX primitive	Parameter syntax	Meaning
\api@xref@rememberthat	**<X> is <Y>; or <X> refers to <Y>**	**defines key and value for references in future and in the past**
\api@xref@referto	**<X>;**	**inserts reference to X**

4.10 RTFMath/MathType equation-building primitives

pseudoTeX primitive	Parameter syntax	Meaning
\mtef@underover@ubar	**<math>**	$\underline{\underline{<math>}}$
\mtef@underover@obar	**<math>**	$\overline{\overline{<math>}}$
\mtef@underover@tilde	**<math>**	$\tilde{<math>}$
\mtef@underover@hat	**<math>**	$\hat{<math>}$
\mtef@underover@arc	**<math>**	$\overset{\frown}{<math>}$
\mtef@underover@jstat	**<math>**	$\overset{\frown}{\overset{\frown}{<math>}}$
\mtef@underover@obar@d	**<math>**	$\overline{\overline{<math>}}$
\mtef@underover@ubar@d	**<math>**	$\underline{\underline{<math>}}$
\mtef@underover@vec	**<math>**	$\overrightarrow{<math>}$
\mtef@underover@vec@l	**<math>**	$\overleftarrow{<math>}$
\mtef@underover@vec@h	**<math>**	$\overrightarrow{<math>}$
\mtef@underover@vec@lr	**<math>**	$\overleftrightarrow{<math>}$
\mtef@underover@vec@u	**<math>**	$\underrightarrow{<math>}$
\mtef@underover@vec@u@l	**<math>**	$\underleftarrow{<math>}$
\mtef@underover@vec@u@h	**<math>**	$\underrightarrow{<math>}$
\mtef@underover@vec@u@lr	**<math>**	$\underleftrightarrow{<math>}$
\mtef@underover@strike@m	**<math>**	$\cancel{<math>}$
\mtef@underover@strike	**<math>**	$\cancel{<math>}$
\mtef@underover@strike@up	**<math>**	$\cancel{<math>}$
\mtef@underover@strike@dn	**<math>**	$\cancel{<math>}$
\mtef@underover@brace@t	**<math>**	$\overbrace{<math>}$
\mtef@underover@brace	**<math>**	$\underbrace{<math>}$
\mtef@underover@brack@t	**<math>**	$\overbracket{<math>}$
\mtef@underover@brack	**<math>**	$\underbracket{<math>}$
\mtef@attachprime	**N/A**	**attaches “prime” mark (or one more “prime” mark) to last element**
\mtef@matrix	**{<cell>&…\cr…}**	**creates matrix**
\mtef@eqno	**<number>**	**marks current subequation number**
\mtef@noeqno	**N/A**	**marks current equation as multiline equation with no number(s)**
\mtef@style	**[=]<N>**	**sets current style: 0 – MATH,1 – TEXT, 2 – FUNCTION, 3 – VECTOR/MATRIX**

4.11 Special primitives

pseudoTeX primitive	Parameter syntax	Meaning
\api@parser@normaldimen	**N/A**	**tosses "dimension" specification**
\api@parser@spec	**N/A**	**tosses "alignment preamble" specification**
\api@parser@glue	**N/A**	**tosses "glue" specification**
\api@parser@rulespec	**N/A**	**tosses "rule" specification**
\api@math@output	**[=]<N>**	**Sets output equation format, <N>:** **0 – MathType v.4+** **1 – Equation Editor 2007/2010**

Appendix A. License Agreement

License Grant

The accompanying Software and User Manual are proprietary products owned by Chikrii Softlab, and protected under U.S. and international copyright law. Except as authorized under this License Grant, the Software may be used only on computers owned, leased, or otherwise controlled by you.

You may not reverse assemble, reverse compile, or otherwise translate the Software.

You may make copies of the software for backup purposes. Except as authorized under this License Grant, no copies of the Software or the User Manual may be made by you or any person under your authority or control.

Use of Software by a commercial organisation for providing document translation services to any third party requires Professional License from Chikrii Softlab if any payment is made in connection with such services.

Limited Warranty

Chikrii Softlab makes no warranties, including warranties of merchantability and fitness for a particular purpose. Any remedy under this warranty is limited to return of the software and the user manual to the dealer or to Chikrii Softlab for replacement or refund. Refund can be performed only before unlock code(s) for Software was (were) generated. This warranty is valid for 90 days from your date of purchase. Chikrii Softlab excludes any warranty coverage for incidental or consequential damages. Some states do not allow limitations on implied warranties, so some of the above limitations may not apply to you.