

**NAME**

`text2troff_table` – converts tabulated lines in a plain-text-file into TROFF formatted tables by inserting table requests.

**SYNOPSIS**

`text2troff_table [options] [input-file]`

**DESCRIPTION**

`text2troff_table` is a bash script that uses *awk* filtering to convert tabulated lines in a plain-text-file into TROFF formatted tables by inserting table requests and ms macros. All lines not interpreted as table lines are passed on unchanged. The result is sent to standard output, and can be processed further with available ‘troff’ tools with ‘ms’ macro package, preferably GNU groff\_ms(7), to produce typeset PostScript-, PDF-, HTML- or terminal (‘nroff’) output. As an alternative to processing a text-file, `text2troff_table` can also read (text) input from a pipe.

**INPUT TEXT REQUIREMENTS**

In order to be interpreted as a table, lines must appear clustered as a continuous series of two (2) or more text lines, each containing at least two (2) non-adjacent *tab* characters, excluding any leading or trailing *tabs* per line. This means that a minimum of three (3) text columns is required to be interpreted as a table.

Any continuous series of one or more *tabs* within a text line is treated as **one** delimiter between adjacent table-‘cells’ c.q. columns. This enables the use of multiple *tabs* in the input text, for instance to align text belonging to the same column for the sake of readability. As a consequence, table cells supposed to be ‘empty’ shall be filled by at least a *space* or e.g. a *dash* (–) or any other chosen character. Another consequence is that a *tab* character itself can – by definition – never be part of the contents of a table cell.

All text intended to be in a single table row shall be placed on the same text line.

**OPTIONS**

`text2troff_table` takes the following options:

- h Print ‘usage’ text for help.
- e Expand tables to full width of the running text margins.  
Without this option, tables are centered with respect to the running text margins and not wider than strictly required by content.
- s Force (‘squeeze’) tables containing too many columns and/or unbreakable text to fit within running text margins, by adding invisible break characters to the contained text.

**TEXT TRANSFORMATION TO TROFF FORMAT**

First, `text2troff_table` removes any amount of *tabs* at the beginning or end of each line, and it replaces any contiguous series of *tabs* within a text line by just one *tab*. Not affected though are text lines within ‘displays’, as marked by presence of ms macros **.DS** (‘display starts’) and **.DE** (‘display ends’). Also all ISO-8859 characters are converted to UTF-8, and *return* characters (\r) are removed.

Next, `text2troff_table` treats any contiguous cluster of two (2) or more text lines as a table, if each contains at least two (2) non-adjacent *tab* characters. The table is terminated as soon as a line passes that doesn’t meet above conditions. At any following cluster of lines that again complies, a next table starts.

Per table, the column quantity is the maximum number of tab-delimited cells found in any of its lines.

The table-number is not hard-coded but instead driven by a TROFF register variable, which is automatically incremented during interpretation by groff\_ms.

Default font size within the tables is set at the default point size minus 20%. Latter value is defined by a second TROFF register variable. To set both variables, `text2troff_table` inserts the following two lines at the top of the input text:

<code>.nr table_nr 0</code>	Initialize new number register ‘table_nr’ at 0
<code>.nr ps_decr \n[.s]*2/10</code>	Set new number register ‘ps_decr’ at 20% of point size

At the top of each table, two lines with ms macro **.LP** (‘paragraph starts’) and **.TS H** (‘table with header starts’) respectively are inserted, as well a number of requests defining the table header and other table

properties.

Per table line, each individual table cell as delimited by adjacent tab(s) is:

- prefixed by `T{` and a *newline* character, and
- postfixed by a *newline* character and `T}`

This formatting allows the text to be wrapped within the cell by breaking at *spaces* and syllable ends when processed by groff\_ms.

Concluding each table, two lines with ms macro `.TE` ('table ends') and `.LP` ('paragraph starts') respectively are inserted.

For example, the original text:

```
After this line of main text, next lines represent a table:
first cell tab second cell: long text tab third cell
fourth cell tab fifth cell tab tab sixth cell
Here the main text continues after the table ...
```

is converted by text2troff\_table the following way:

After this line of main text, next lines represent a table:	
<code>.LP</code>	Macro 'paragraph starts'
<code>.nr table_nr +1</code>	Number register 'table_nr' is incremented by 1
<code>.ps -\n[ps_decr]</code>	Point size minus decrement value being 20% of point size
<code>.TS H</code>	Request 'table with header starts'
<code>Tabel \n[table_nr]</code>	Header text and incremented number register 'table_nr'
<code>table_type, allbox;</code>	Requests 'table_type' and 'box around every table cell'
<code>c s s</code>	Header cell centered (c), spans (s) each other column
<code>l l l .</code>	Each cell of next rows left (l) aligned
<code>T{</code>	
first cell	First row of table cells
<code>T} tab T{</code>	
second cell: long text	
<code>T} tab T{</code>	
third cell	
<code>T}</code>	
<code>T{</code>	
fourth cell	Second row of table cells
<code>T} tab T{</code>	
fifth cell	
<code>T} tab T{</code>	
sixth cell	
<code>T}</code>	
<code>.TE</code>	Request 'table ends'
<code>.LP</code>	ms macro 'paragraph starts'
Here the main text continues after the table ...	

The `table_type` can be:

`center`

In this default case, groff\_ms will center the table between the text margins, at a width no more than strictly required by table contents;

`expand`

if option `-e` is given, or if option `-s` is given and table exceeds text margins, groff\_ms will expand the table to full width between the text margins.

if option `-s` is given, invisible break characters `\:` are inserted in the table cell text after every *n* characters –

with  $n$  depending on on table width, chosen number of columns and character type and size – as to have groff\_ms break the lines and ‘squeeze’ the table between text margins.

## COMPATIBILITY

This program has been tested to run on both GNU/Linux (Ubuntu 22.04, Tiny Core Linux 10, Alpine Linux 3.12 within iSH app on iOS smartphone), as well as on MacOS X.

## BUGS

No known bugs.

## AUTHOR

Written by Rob Toscani (rob\_toscani@yahoo.com)

## REPORTING BUGS

Please report any bugs to the author by e-mail or via <https://github.com/jazzfan2/text2troff/issues>