

The `lengthconvert` package

Marco Daniel E-mail: marco.daniel@mada-nada.de

Released 2013/06/13

Sometimes it's useful for any explanation to provide length in standardizations units. This package can do this for your.

The command is needed for one of my other packages so I decided to write a small package.

Contents

1	Basics	1
2	Usage	2
3	Options	2
4	Examples	3
5	<code>lengthconvert</code> Implementation	4
	Change History	10
	Index	11

1 Basics

The package needs the newest version of `l3kernel` available at CTAN. Internally it uses the modul `l3fp` to convert the length.

All allowed length are listed in the table below.

Table 1: Allow T_EX units

Unit	Measurement
pt	point
pc	pica(1pc=12pt)
in	inch (1 in = 72.27 pt)
bp	bigpoint(72bp=1in)
cm	centimeter (2.54 cm = 1 in)
mm	millimeter (10 mm = 1 cm)
dd	didot point (1157 dd = 1238 pt)
cc	cicero (1 cc = 12 dd)
sp	scaled point (65536 sp = 1 pt)

2 Usage

The usage is really simple. At the length to the main command and get the result.

\Convert `\Convert[<options>] {<length>}`

The command converts the given length to the unit specified by an option. After the conversion the result will be printed.

\Convertsetup `\Convertsetup {<options>}`

Allows the specification of options.

3 Options

The package is simple and the options too.

unit The option accept only the abbreviation unit. Allowed units are described in the table above. You can also use only the abbreviation or complete word phrases without the option. The following table list all allowed inputs.

		pt	pc	in	bp	cm	mm
		dd	cc	sp	point	pica	inch
		big-point	centimeter	millimeter	didot-point	cicero	scaled-point
use-siunitx	It's a bool flag which can be either true or false . If it is true, the output of the new length is done by the package siunitx using the command <code>\SI</code> .						
precision	This option accepts an integer and specifies the precision of the output.						
number-only	It's a bool flag which can be either true or false . If it's true, only the number is printed.						

4 Examples

Some examples are shown in the following table. In the left column you see the input and in the right the output.

<code>\Convert{36pt}</code>	1.26526 cm
<code>\Convert[precision=2]{36pt}</code>	1.27 cm
<code>\Convert[use-siunitx]{36pt}</code>	1.265 26 cm
<code>\Convert[unit=pt]{2cm}</code>	56.9055 pt
<code>\Convert[unit=dd,number-only]{2cm}</code>	53.18229
<code>\Convert[pt]{2cm}</code>	56.9055 pt
<code>\Convert[scaled-point]{2cm}</code>	3729359 sp

5 lengthconvert Implementation

```

1 <*package>
2 <@@=lconv>
3 \ProvidesExplPackage
4   {lengthtconvert}{2013/05/13}{1.0}{Convert length to another unit}

```

Make sure that the version of l3kernel in use is sufficiently new. This will also trap any problems with l3packages (as the two are now tied together, version-wise).

```

5 \IfpackageLater { expl3 } { 2012/11/21 }
6   { }
7   {
8     \PackageError { lengthtconvert } { Support~package~expl3~too~old }
9     {
10       You~need~to~update~your~installation~of~the~bundles~'l3kernel'~and~
11       'l3packages'.\MessageBreak
12       Loading~lengthtconvert~will~abort!
13     }
14     \tex_endinput:D
15   }

```

Now load the support packages.

```

16 \RequirePackage{ l3keys2e }

```

`_lconv_allowed_shortunits_clist` Save all allowed units in a clist

`_lconv_allowed_longunits_clist`
`_lconv_allowed_allunits_clist`

```

17 \clist_new:N \l__lconv_allowed_shortunits_clist
18 \clist_set:Nn \l__lconv_allowed_shortunits_clist
19   { pt , pc , in , bp , cm , mm , dd , cc ,sp }
20 \clist_new:N \l__lconv_allowed_longunits_clist
21 \clist_set:Nn \l__lconv_allowed_longunits_clist
22   { point , pica , inch , big-point , centimeter , millimeter ,
23     didot-point , cicero , scaled-point }
24 \clist_new:N \__lconv_allowed_allunits_clist
25 \clist_set:NV \__lconv_allowed_allunits_clist \l__lconv_allowed_shortunits_clist
26 \clist_put_right:NV \__lconv_allowed_allunits_clist \l__lconv_allowed_longunits_clist

```

(End definition for `_lconv_allowed_shortunits_clist`. This function is documented on page ??.)

`_lconv_unit_tl` Save the default unit in a token list variable and provide them as option

```

27 \tl_new:N \l__lconv_unit_tl
28 \keys_define:nn { lengthconvert }
29 {
30   unit .tl_set:N = \l__lconv_unit_tl
31 }
32 \keys_set:nn { lengthconvert }
33 {
34   unit = cm ,
35 }

```

Provide also abbreviation and word of units

```

36 \tl_new:N \l__lconv_default_unit_tl
37 \keys_define:nn { lengthconvert }
38 {
39   pt .meta:n =
40     { unit = pt },
41   pc .meta:n =
42     { unit = pc },
43   in .meta:n =
44     { unit = in },
45   bp .meta:n =
46     { unit = bp },
47   cm .meta:n =
48     { unit = cm },
49   mm .meta:n =
50     { unit = mm },
51   dd .meta:n =
52     { unit = dd },
53   cc .meta:n =
54     { unit = cc },
55   sp .meta:n =
56     { unit = sp },
57   point .meta:n =
58     { unit = pt },
59   pica .meta:n =
60     { unit = pc },
61   inch .meta:n =
62     { unit = in },

```

```

63     big-point .meta:n =
64         { unit = bp },
65     centimeter .meta:n =
66         { unit = cm },
67     millimeter .meta:n =
68         { unit = mm },
69     didot-point .meta:n =
70         { unit = dd },
71     cicero .meta:n =
72         { unit = cc },
73     scaled-point .meta:n =
74         { unit = sp },
75 }

```

(End definition for `_lconv_unit_tl`. This function is documented on page ??.)

`\l__lconv_use_siunitx_bool` Output should be done by sinutix.

```

76 \keys_define:nn { lengthconvert } {
77     use-siunitx .bool_set:N = \l__lconv_use_siunitx_bool
78 }

```

(End definition for `\l__lconv_use_siunitx_bool`. This function is documented on page ??.)

`\l_lconv_precision_tl` Specify the precision

```

79 \keys_define:nn { lengthconvert } {
80     precision .tl_set:N = \l_lconv_precision_tl
81 }
82 \keys_set:nn { lengthconvert }
83 {
84     precision = 5 ,
85 }

```

(End definition for `\l_lconv_precision_tl`. This function is documented on page ??.)

`\l__lconv_only_num_bool` Only the number should be used

```

86 \keys_define:nn { lengthconvert } {
87     number-only .bool_set:N = \l__lconv_only_num_bool
88 }

```

(End definition for `\l__lconv_only_num_bool`. This function is documented on page ??.)

Unknown options should be raised an error

```

89 \keys_define:nn { lengthconvert } {
90   unknown .code:n =
91   {
92     \msg_error:nnx { lengthconvert } { option-unknown }
93     { \exp_not:V \l_keys_key_tl }
94   }
95 }

```

\Convertsetup User settings

```

96 \NewDocumentCommand \Convertsetup { m }
97 {
98   \keys_set:nn { lengthconvert } { #1 }
99 }

```

(End definition for `\Convertsetup`. This function is documented on page 2.)

\Convert Expandable definition of the main command

```

100 \DeclareExpandableDocumentCommand \Convert { 0{} m }
101 {
102   \group_begin:
103     \keys_set:nn { lengthconvert } { #1 }
104     \clist_if_in:NVTf \__lconv_allowed_allunits_clist \l__lconv_unit_tl
105     {
106       \bool_if:NTF \l__lconv_use_siunitx_bool
107       {
108         \__lconv_using_siunitx:n { #2 }
109       }
110       {
111         \__lconv_nousing_siunitx:n { #2 }
112       }
113     }
114     {
115       \msg_error:nnx { lengthconvert } { unit-unknown }
116       { \exp_not:V \l__lconv_unit_tl }
117     }
118   \group_end:

```

```
119 }
```

(End definition for \Convert. This function is documented on page 2.)

`__lconv_using_siunitx` Output using siunitx

```
120 \cs_new:Npn \__lconv_using_siunitx:n #1
121 {
122   \bool_if:NTF \l__lconv_only_num_bool
123   {
124     \num
125     {
126       \fp_eval:n
127       {
128         round( \dim_to_fp:n { #1 } / 1\l__lconv_unit_tl , \l__lconv_precision_tl)
129       }
130     }
131   }
132   {
133     \SI
134     {
135       \fp_eval:n
136       {
137         round( \dim_to_fp:n { #1 } / 1\l__lconv_unit_tl , \l__lconv_precision_tl)
138       }
139     }
140     {
141       \l__lconv_unit_tl
142     }
143   }
144 }
```

(End definition for __lconv_using_siunitx. This function is documented on page ??.)

`__lconv_nousing_siunitx` Output using siunitx

```
145 \cs_new:Npn \__lconv_nousing_siunitx:n #1
146 {
147   \bool_if:NTF \l__lconv_only_num_bool
148   {
149     \fp_eval:n
150     {
```



```

151     round( \dim_to_fp:n { #1 } / 1\l___lconv_unit_tl , \l___lconv_precision_tl)
152   }
153 }
154 {
155   \fp_eval:n
156   {
157     round( \dim_to_fp:n { #1 } / 1\l___lconv_unit_tl , \l___lconv_precision_tl)
158   } \,
159   \l___lconv_unit_tl
160 }
161 }

```

(End definition for _lconv_nousing_siunitx. This function is documented on page ??.)

```

162 \msg_new:nnnn { lengthconvert } { option-unknown }
163   { Unknown-option-’#1’-for-package-#2. }
164   {
165     LaTeX-has-been-asked-to-set-an-option-called-’#1’-
166     but-the-#2-package-has-not-created-an-option-with-this-name.
167   }

168 \msg_new:nnnn { lengthconvert } { unit-unknown }
169   { Unknown-unit-’#1’-for-package-#2. }
170   {
171     You-are-setting-an-unit-’#1’-which-
172     is-unknown-for-the-package-#2.
173   }

```

Finally apply the settings given at load time.

```

174 \ProcessKeysOptions { lengthconvert }

175 </package>

```

Change History

v1.0

General: First official release 1

Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

Symbols	G
\, 158	\group_begin: 102
\@ifpackagelater 5	\group_end: 118
_lconv_allowed_allunits_clist	
..... 17, 24, 25, 26, 104	
_lconv_allowed_longunits_clist ... 17	
_lconv_allowed_shortunits_clist .. 17	
_lconv_nousing_siunitx 145	
_lconv_nousing_siunitx:n 111, 145	
_lconv_unit_tl 27	
_lconv_using_siunitx 120	
_lconv_using_siunitx:n 108, 120	
B	K
\bool_if:NTF 106, 122, 147	\keys_define:nn 28, 37, 76, 79, 86, 89
	\keys_set:nn 32, 82, 98, 103
C	L
\clist_if_in:NVTF 104	\l__lconv_allowed_longunits_clist .
\clist_new:N 17, 20, 24 20, 21, 26
\clist_put_right:NV 26	\l__lconv_allowed_shortunits_clist
\clist_set:Nn 18, 21 17, 18, 25
\clist_set:NV 25	\l__lconv_only_num_bool 86, 87, 122, 147
\Convert 2, 100, 100	\l__lconv_precision_tl
\Convertsetup 2, 96, 96 80, 128, 137, 151, 157
\cs_new:Npn 120, 145	\l__lconv_unit_tl 27, 30,
	104, 116, 128, 137, 141, 151, 157, 159
	\l__lconv_use_siunitx_bool . 76, 77, 106
	\l__lconv_default_unit_tl 36
	\l__lconv_precision_tl 79
	\l_keys_key_tl 93
D	M
\DeclareExpandableDocumentCommand . 100	\MessageBreak 11
\dim_to_fp:n 128, 137, 151, 157	\msg_error:nnx 92, 115
	\msg_new:nnnn 162, 168
E	N
\exp_not:V 93, 116	\NewDocumentCommand 96
	\num 124
	number-only (option) 3
F	O
\fp_eval:n 126, 135, 149, 155	options:
	number-only 3

precision	3		S
unit	2	\SI	133
use-siunitx	3		
P		T	
\PackageError	8	\tex_endinput:D	14
precision (option)	3	\tl_new:N	27, 36
\ProcessKeysOptions	174		
\ProvidesExplPackage	3		U
R		unit (option)	2
\RequirePackage	16	use-siunitx (option)	3