

# The `lengthconvert` package

Marco Daniel E-mail: [marco.daniel@mada-nada.de](mailto:marco.daniel@mada-nada.de)

Released 2013/06/13

Sometimes it's useful for some explanation to provide lengths in standardizations units instead of the default unit of  $\TeX$ . This package can do this for your.

## Contents

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

## 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 units in  $\TeX$  are listed in the table below.

Table 1: Allowed T<sub>E</sub>X 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. Pass the length to the command `\Convert` and get the result.

---

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

---

The command converts the given length to the unit specified by an option. The default unit is cm. 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 accepts only the abbreviation unit. Allowed units are described in the table above.

You can also use only the abbreviation or a complete word. The following table lists 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 `\SI`.

**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   {lengthconvert}{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 \g___lconv_allowed_shortunits_clist
18 \clist_gset:Nn \g___lconv_allowed_shortunits_clist
19   { pt , pc , in , bp , cm , mm , dd , cc ,sp }
20 \clist_new:N \g___lconv_allowed_longunits_clist
21 \clist_gset:Nn \g___lconv_allowed_longunits_clist
22   { point , pica , inch , big-point , centimeter , millimeter ,
23     didot-point , cicero , scaled-point }
24 \clist_new:N \g___lconv_allowed_allunits_clist
25 \clist_gset:NV \g___lconv_allowed_allunits_clist \g___lconv_allowed_shortunits_clist
26 \clist_gput_right:NV \g___lconv_allowed_allunits_clist \g___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 .int_set:N = \l__lconv_precision_int
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 \g__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_calc_dim:n` Output using siunitx

```
120 \cs_new:Npn \_lconv_calc_dim:n #1
121 {
122   \fp_eval:n
123     {
124       round( \dim_to_fp:n { #1 } / 1\l__lconv_unit_tl , \l__lconv_precision_int)
125     }
126 }
```

*(End definition for \\_lconv\_calc\_dim:n. This function is documented on page ??.)*

`\_lconv_using_siunitx:n` Output using siunitx

```
127 \cs_new:Npn \_lconv_using_siunitx:n #1
128 {
129   \bool_if:NTF \l__lconv_only_num_bool
130     {
131       \num { \_lconv_calc_dim:n { #1 } }
132     }
133     {
134       \SI { \_lconv_calc_dim:n { #1 } } { \l__lconv_unit_tl }
135     }
136 }
```

*(End definition for \\_lconv\_using\_siunitx:n. This function is documented on page ??.)*

`\_lconv_nousing_siunitx:n` Output using siunitx

```
137 \cs_new:Npn \_lconv_nousing_siunitx:n #1
138 {
139   \bool_if:NTF \l__lconv_only_num_bool
140     {
141       \_lconv_calc_dim:n { #1 }
142     }
143     {
144       \_lconv_calc_dim:n { #1 } \, \l__lconv_unit_tl
145     }
146 }
```



*(End definition for `\_lconv_nousing_siunitx:n`. This function is documented on page ??.)*

```
147 \msg_new:nnnn { lengthconvert } { option-unknown }
148   { Unknown-option-’#1’-for-package-#2. }
149   {
150     LaTeX-has-been-asked-to-set-an-option-called-’#1’-
151     but-the-#2-package-has-not-created-an-option-with-this-name.
152   }

153 \msg_new:nnnn { lengthconvert } { unit-unknown }
154   { Unknown-unit-’#1’-for-package-#2. }
155   {
156     You-are-setting-an-unit-’#1’-which-
157     is-unknown-for-the-package-#2.
158   }
```

Finally apply the settings given at load time.

```
159 \ProcessKeysOptions { lengthconvert }

160 </package>
```

## Change History

v1.0		v1.0a	
General: First official release . . . . .	1	General: fixed typo in package name . . . . .	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
<code>\,</code> ..... 144	<code>\g__lconv_allowed_allunits_clist</code> .. ..... 24, 25, 26
<code>\@ifpackagelater</code> ..... 5	<code>\g__lconv_allowed_longunits_clist</code> . ..... 20, 21, 26
<code>\_lconv_allowed_allunits_clist</code> .... <u>17</u>	<code>\g__lconv_allowed_shortunits_clist</code> ..... 17, 18, 25
<code>\_lconv_allowed_longunits_clist</code> .... <u>17</u>	<code>\g__lconv_allowed_allunits_clist</code> .. 104
<code>\_lconv_allowed_shortunits_clist</code> .. <u>17</u>	<code>\group_begin:</code> ..... 102
<code>\_lconv_calc_dim:n</code> ..... ..... <u>120</u> , <u>120</u> , <u>131</u> , <u>134</u> , <u>141</u> , <u>144</u>	<code>\group_end:</code> ..... 118
<code>\_lconv_nousing_siunitx:n</code> 111, <u>137</u> , <u>137</u>	
<code>\_lconv_unit_tl</code> ..... <u>27</u>	
<code>\_lconv_using_siunitx:n</code> .. 108, <u>127</u> , <u>127</u>	
	<b>K</b>
<b>B</b>	<code>\keys_define:nn</code> .... 28, 37, 76, 79, 86, 89
<code>\bool_if:NTF</code> ..... 106, 129, 139	<code>\keys_set:nn</code> ..... 32, 82, 98, 103
	<b>L</b>
<b>C</b>	<code>\l__lconv_only_num_bool</code> 86, 87, 129, 139
<code>\clist_gput_right:NV</code> ..... 26	<code>\l__lconv_precision_int</code> ..... 80, 124
<code>\clist_gset:Nn</code> ..... 18, 21	<code>\l__lconv_unit_tl</code> ..... ..... 27, 30, 104, 116, 124, 134, 144
<code>\clist_gset:NV</code> ..... 25	<code>\l__lconv_use_siunitx_bool</code> . <u>76</u> , <u>77</u> , 106
<code>\clist_if_in:NVTF</code> ..... 104	<code>\l_lconv_default_unit_tl</code> ..... 36
<code>\clist_new:N</code> ..... 17, 20, 24	<code>\l_lconv_precision_tl</code> ..... <u>79</u>
<code>\Convert</code> ..... 2, <u>100</u> , 100	<code>\l_keys_key_tl</code> ..... 93
<code>\Convertsetup</code> ..... 2, <u>96</u> , 96	
<code>\cs_new:Npn</code> ..... 120, 127, 137	
	<b>M</b>
<b>D</b>	<code>\MessageBreak</code> ..... 11
<code>\DeclareExpandableDocumentCommand</code> . 100	<code>\msg_error:nnx</code> ..... 92, 115
<code>\dim_to_fp:n</code> ..... 124	<code>\msg_new:nnnn</code> ..... 147, 153
	<b>N</b>
<b>E</b>	<code>\NewDocumentCommand</code> ..... 96
<code>\exp_not:V</code> ..... 93, 116	<code>\num</code> ..... 131
	number-only (option) ..... 2
<b>F</b>	
<code>\fp_eval:n</code> ..... 122	

	<b>O</b>		<b>R</b>
options:		<code>\RequirePackage</code> .....	16
number-only .....	2		<b>S</b>
precision .....	2	<code>\SI</code> .....	134
unit .....	2		<b>T</b>
use-siunitx .....	2	<code>\tex_endinput:D</code> .....	14
	<b>P</b>	<code>\tl_new:N</code> .....	27, 36
<code>\PackageError</code> .....	8		<b>U</b>
precision (option) .....	2	unit (option) .....	2
<code>\ProcessKeysOptions</code> .....	159	use-siunitx (option) .....	2
<code>\ProvidesExplPackage</code> .....	3		