

The docext Package^{*}

Mingyu Xia[†]

Released 2026-06-27 v0.2B

Abstract

The docext package is an extension for documenting the L^AT_EX source files. It is based on the doc package and works smoothly under the l₃doc, ltxdoc, and article classes.

1 Loading & Configuring the Package

Write the line in the preamble

```
\usepackage[options]{docext}
```

while the package $\langle options \rangle$ accepts the following keys

`color` = $\langle 13color | HTML \rangle$: The color of keys (Default: black).

`font` = $\langle font macros \rangle$: Font of the keys.

`delimiter` = $\langle string \rangle$: Delimiter between the keys.

`ratio` = $\langle fp num \rangle$: Ratio of the boxes' widths in codemo environment (Default: 0.625).

Apart from configuring the package with the package options, users can also use the `\docextset` function within the preamble and the context.

<code>\docextset</code>	<code>\docextset {\langle options \rangle}</code>
-------------------------	---

<code>New: 2026-06-07</code>	The functions used in this package <i>after</i> <code>\docextset</code> will be influenced. The $\langle options \rangle$ accepts the same keys above.
------------------------------	---

^{*}<https://ctan.org/pkg/docext>, <https://github.com/myhsia/docext>

[†]xiamingyu@westlake.edu.cn

2 User's Interfaces

`\keys` `\keys [<options>] {<comma list>}`

New: 2026-06-07 Typeset the keys with specific format. The `<options>` accepts the `delimiter`, `font`,
Updated: 2026-06-27 `color` keys referred in [section 1](#) and another `emph` key to assign # of the key will be emphasised.

The initial value of the `delimiter` key is `,\,\allowbreak` in this function.

<pre>\keys [font = \sl] {keya, keyb, keyc}\\ \keys [emph = 2, delimiter = \textbar] {true, false, not given}</pre>	<pre>keya, keyb, keyc true false not given</pre>
--	--

`\val` `\val [<options>] {<comma list>}`

New: 2026-06-13 Typeset the value with specific format and surround with `\meta`. The `<options>` accepts
the `delimiter`, `font`, `color` keys referred in [section 1](#).

The initial value of the `delimiter` key is `\textbar` in this function.

`\TF` Shortcut for typeset Boolean-choice value.

New: 2026-06-07

<code>\TF, \TTF, \TFF.</code>	<code>true false, true false, true false.</code>
-------------------------------	--

`\ttwd` `\hspace \<fontsize>ttwd`

New: 2026-06-13 Output the dimension of the width of one typewriter letter. The `<fontsize>` prefix
can be chosen as `tiny`, `scriptsize`, `footnotesize`, `small`, `normalsize`, `large`,
`Large`, `LARGE`, `huge`, `Huge`.

`codemo`, `codemo* (env)` Typeset the code on the left hand side and then execute the code in the right hand side,
while the star version will put the code above the demo.

```
\begin{codemo(*)}
Code needs to be typeset in verbatim and demoed
\end{codemo(*)}
```

`keyval (env)` The effect is the same as the demos in the introductions of the functions above. Itemize-
type environment for typesetting the keys.

```
\begin{keyval}
\item [\keys{keya, keyb, keyc}]\meta{foo}
\end{keyval}
```

The effect is the same as the package options introduction of [section 1](#) in this manual.

A The source code

Using codedoc as the namespace.

```
1 <@@=codedoc>
```

Start the optionlist package for l3docstrip.

```
2 <*package>
```

Key-value definitions of the package options, separate the keys into two groups. The first one, @@ / pkgopt / keys, is also used in \keys.

```
__color_named_docolor_prop 3 \keys_define:nn { __codedoc / pkgopt / keys }
\g__codedoc_keys_font_tl    4 {
\g__codedoc_keys_delimiter_tl 5   color      .code:n =
g__codedoc_codemo_ratio_fp 6   {
7     \color_if_exist:nTF {#1}
8     { \color_set:nn { docolor } {#1} }
9     { \color_set:nnn { docolor } { HTML } {#1} }
10  },
11  color      .initial:n = { black },
12  font       .tl_set:N = \g__codedoc_keys_font_tl,
13  delimiter  .tl_set:N = \g__codedoc_keys_delimiter_tl,
14  }
15 \keys_define:nn { __codedoc / pkgopt / codemo }
16 {
17   ratio      .fp_set:N = \g__codedoc_codemo_ratio_fp,
18   ratio      .initial:n = { .625 },
19 }
```

(End of definition for \l__color_named_docolor_prop and others.)

Process the class options.

```
20 \ProcessKeyOptions [ __codedoc / pkgopt / keys,
21                    __codedoc / pkgopt / codemo ]
```

\docextset To control the variables.

```
22 \DeclareDocumentCommand \docextset { m }
23 {
24   \keys_set_known:nn { __codedoc / pkgopt / keys } {#1}
25   \keys_set_known:nn { __codedoc / pkgopt / codemo } {#1}
26 }
```

(End of definition for \docextset. This function is documented on page 1.)

Load the dependencies.

```
27 \RequirePackage { xcolor, enumitem, verbatim }
```

Overwrite `\prepare@family@series@update` again under `pLATEX` and `upLATEX` to resolve the error raised by `\settowidth \MacroIndent` in the doc package.

```

28 \bool_lazy_or:nnT { \sys_if_engine_ptex_p: } { \sys_if_engine_uptex_p: }
29 {
30   \def\prepare@family@series@update#1#2{%
31     \def\@currentmetafamily{#1}%
32     \if@forced@series
33       \fontfamily#2%
34     \else
35       \expand@font@defaults
36       \let\@target@series@value\@empty
37       \expandafter\edef\csname ??def@ult\endcsname{\f@family}%
38       \let\@elt@update@series@target@value
39         \@meta@family@list
40         \@elt{??}%
41       \let\@elt\relax
42       \fontfamily#2%
43       \ifx\@target@series@value\@empty
44       \else
45         \ifx \f@series\@target@series@value
46         \else
47           \maybe@load@fontshape
48           \series@maybe@drop@one@m\@target@series@value\f@series
49         \fi
50       \fi
51     \fi
52   }
53 }

```

Patch the font of the macrocode environment.

```

54 \hook_gput_code:nnn { cmd / macro@font / after } { . } { \ttfamily }

```

Patch the macro `\meta` in the `l3doc` class and the doc package. Since the `l3doc` package has already loaded the doc package, a branch is required.

```

55 \cs_if_exist:cTF { ver@l3doc.cls }

```

Under the `l3doc` class, surround `\hbox:n` to the argument of the kernel function `__codedoc_meta_original:n` of the `\meta` macro. Suppress the warning raised by `l3doc` (checking `l3syntax`), and reset the counter: section at the beginning of the documentation environment.

```

56 {
57   \cs_set_eq:NN \__codedoc_meta_patch:n \__codedoc_meta_original:n
58   \cs_set_protected:Npn \__codedoc_meta_original:n #1
59     { \__codedoc_meta_patch:n { \hbox:n {#1} } }
60   \msg_redirect_module:nnn { l3doc } { warning } { none }

```

```

61 \hook_gput_code:nnn { env / documentation / before } { . }
62 { \setcounter { section } { 0 } }
63 }

```

Under other classes, load the doc package then take the same trick as in the l3doc class.

```

64 {
65 \RequirePackage { doc }
66 \cs_set_eq:NN \__codedoc_meta_patch:n \meta
67 \DeclareDocumentCommand \meta { m }
68 { \__codedoc_meta_patch:n { \hbox:n {#1} } }
69 }

```

\keys L^AT_EX 2_ε user's interface for input keys quickly.

Variables for storing the output of __codedoc_keys_aux:nnnnn.

```

\__codedoc_keys_typeset_tl
\__codedoc_keys_typeset_box
70 \tl_new:N \l__codedoc_keys_typeset_tl
71 \box_new:N \l__codedoc_keys_typeset_box
72 \msg_new:nnn { docext } { Overfull ~ \string \keys }
73 {
74 Keys ~ wider ~ than ~ 1.2 \string \marginparwidth \iow_newline:
75 \space\space Exceed ~ the ~ ability ~ of ~ scaling
76 }
77 \DeclareDocumentCommand \keys { 0 {} m }
78 {
79 \group_begin:
80 \keys_set_known:nn { __codedoc / pkgopt / keys }
81 { delimiter = { ,\,\allowbreak }, #1 }
82 \keys_set_known:nn { __codedoc / keys } {#1}
83 \tl_set:Nn \l__codedoc_keys_typeset_tl
84 {
85 \__codedoc_keys_aux:nnnnn {#2}
86 { \l__codedoc_keys_emph_int }
87 { \textup \g__codedoc_keys_delimiter_tl }
88 { \g__codedoc_keys_font_tl \ttfamily }
89 { docolor }
90 }
91 \hbox_set:Nn \l__codedoc_keys_typeset_box { \l__codedoc_keys_typeset_tl }
92 \tl_if_eq:NnTF \@currentvir { keyval }
93 {
94 \dim_compare:nNnT
95 { \box_wd:N \l__codedoc_keys_typeset_box } > { \marginparwidth }
96 {
97 \dim_compare:nNnT
98 { \box_wd:N \l__codedoc_keys_typeset_box } > { 1.2 \marginparwidth }
99 { \msg_warning:nn { docext } { Overfull ~ \string \keys } }

```

```

100         \box_scale:Nnn \l__codedoc_keys_typeset_box { 5 / 6 } { 1 }
101     }
102     \box_use_drop:N \l__codedoc_keys_typeset_box
103     \g__codedoc_keys_font_tl \ttfamily \, = \,
104     } { \l__codedoc_keys_typeset_tl }
105 \group_end:
106 }

```

Key-value definitions special for the `\keys` macro.

```

\l__codedoc_keys_emph_int 107 \keys_define:nn { __codedoc / keys }
108   { emph .int_set:N = \l__codedoc_keys_emph_int }

```

(End of definition for `\keys` and others. This function is documented on page 2.)

\val ~~LaTeX~~ 2ϵ user's interface for input value quickly.

```

109 \DeclareDocumentCommand \val { 0 {} m }
110 {
111   \group_begin:
112     \keys_set_known:nn { __codedoc / pkgopt / keys }
113     { delimiter = { \g__codedoc_vbar_tl \allowbreak }, #1 }
114     \meta
115     {
116       \__codedoc_keys_aux:nnnnn {#2}
117       { 0 }
118       { \textup \g__codedoc_keys_delimiter_tl }
119       { \g__codedoc_keys_font_tl \ttfamily }
120       { docolor }
121     }
122   \group_end:
123 }

```

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

`__codedoc_keys_aux:nnnnn` Auxiliary commands of the `\keys` macro.

`__codedoc_keys_emph:n`

#1: comma list, #4: font,
 #2: # of the element to be emphasised, #5: color (works only for the key),
 #3: delimiter,

```

124 \cs_new_protected_nopar:Npn \__codedoc_keys_aux:nnnnn #1#2#3#4#5
125 {
126   \group_begin:
127     \seq_set_from_clist:Nn \l_tmpa_seq {#1}
128     \int_if_zero:nF {#2}
129     {

```

```

130         \seq_gset_item:Nnn \l_tmpa_seq {#2}
131         { \__codedoc_keys_emph:n { \clist_item:nn {#1} {#2} } }
132     }
133     \seq_set_map:Nnn \l_tmpb_seq \l_tmpa_seq
134     {
135         \color_group_begin:
136         \exp_args:No \color_select:n {#5} ##1
137         \color_group_end:
138     }
139     #4 \seq_use:Nn \l_tmpb_seq {#3}
140 \group_end:
141 }
142 \cs_new_protected_nopar:Npn \__codedoc_keys_emph:n #1
143 {
144     \group_begin:
145     \upshape \bfseries \exp_args:No \color_select:n { red } #1
146     \group_end:
147 }

```

(End of definition for `__codedoc_keys_aux:nnnnn` and `__codedoc_keys_emph:n`.)

\TF \LaTeX 2 ϵ user's interface for input Boolean-choice value quickly.

```

\TTF 148 \DeclareDocumentCommand \TF {} { \__codedoc_keys_TF_aux:n { 0 } }
\TTF 149 \DeclareDocumentCommand \TTF {} { \__codedoc_keys_TF_aux:n { 1 } }
\TTF 150 \DeclareDocumentCommand \TFF {} { \__codedoc_keys_TF_aux:n { 2 } }

```

(End of definition for `\TF`, `\TTF`, and `\TFF`. These functions are documented on page 2.)

`__codedoc_keys_TF_aux:n` Auxiliary commands of the `\TF`, `\TTF`, `\TFF` macros.

```

151 \cs_new_protected_nopar:Npn \__codedoc_keys_TF_aux:n #1
152 {
153     \__codedoc_keys_aux:nnnnn
154     { \g__codedoc_true_tl, \g__codedoc_false_tl } {#1}
155     { \textup \g__codedoc_vbar_tl } { \ttfamily } { docolor }
156 }

```

Constant variables used in `__codedoc_keys_TF_aux:n`.

```

\g__codedoc_true_tl 157 \tl_const:Nn \g__codedoc_true_tl { true }
\g__codedoc_false_tl 158 \tl_const:Nn \g__codedoc_false_tl { false }
\g__codedoc_vbar_tl 159 \tl_const:Nn \g__codedoc_vbar_tl
160 { \sys_if_engine_opentype:TF { \textbar } { | } }

```

(End of definition for `__codedoc_keys_TF_aux:n` and others.)

keyval (*env.*) Itemize-type environment for typesetting the keys. Using the interfaces provided by the `enumitem` package.

```

161 \newlist{keyval}{itemize}{10}
162 \setlist{keyval}{leftmargin = 0pt, labelsep = 0pt}

```

codemo, codemo* (*env.*) Environment for typesetting the code within a verbatim and a demo environment at the same time.

```

163 \DeclareDocumentEnvironment { codemo } { } { \__codedoc_codemo_start: }
164 {
165     \__codedoc_codemo_end:Nn \c_false_bool
166     { \g__codedoc_codemo_ratio_fp }
167 }
168 \DeclareDocumentEnvironment { codemo* } { } { \__codedoc_codemo_start: }
169 {
170     \__codedoc_codemo_end:Nn \c_true_bool
171     { \g__codedoc_codemo_ratio_fp }
172 }

```

__codedoc_codemo_start: Auxiliary commands of the codemo environment.

__codedoc_codemo_end:Nn 173 \cs_new_protected:Npn __codedoc_codemo_start:
174 {

The I/O streams part¹.

```

175     \group_begin:
176     \@bsphack
177     \iow_open:Nn \l__codedoc_codemo_iow { \l__codedoc_codemo_name_tl }
178     \let \do \char_set_catcode_other:N
179     \dospecials \char_set_catcode_active:N \^^M
180     \cs_set:Npn \verbatim@processline
181     {
182         \exp_args:NNo \iow_now:Nn
183         \l__codedoc_codemo_iow { \int_use:N \verbatim@line }
184     }
185     \verbatim@start
186 }
187 \cs_new_protected:Npn \__codedoc_codemo_end:Nn #1#2
188 {
189     \iow_close:N \l__codedoc_codemo_iow
190     \@esphack
191     \group_end:
192     \char_set_catcode_comment:N \%
193     \ior_open:Nn \l__codedoc_codemo_ior { \l__codedoc_codemo_name_tl }
194     \int_zero:N \l__codedoc_codemo_int
195     \ior_map_inline:Nn \l__codedoc_codemo_ior
196     { \int_incr:N \l__codedoc_codemo_int }

```

¹By modifying @AlphaZTX's [phy2docdef.tex](#) magically using LaTeX3.

The typeset part.

```

197   \par \smallskip
198   \group_begin:
199     \dim_set:Nn \parindent { \c_zero_skip }
200     \fbox
201     {
202       \bool_if:NT #1 { \null \skip_horizontal:n { 2\ttwd } }
203       \parbox [ c ] [ \l__codedoc_codemo_int \baselineskip + .48ex ]
204       {
205         \fp_eval:n { \bool_if:NTF #1 { 1 } { #2 } * .96 }
206         \linewidth \bool_if:NT #1 { -.8pt }
207       } { \small \verbatiminput \l__codedoc_codemo_name_tl }
208     }
209     \bool_if:NTF #1
210     { \par \skip_vertical:n { -1.5pt } } { \hfill }
211     \fbox
212     {
213       \null \skip_horizontal:n { 2\ttwd }
214       \parbox [ c ] [ \l__codedoc_codemo_int \baselineskip + .48ex ]
215       {
216         \fp_eval:n { \bool_if:NTF #1 { 1 } { (1 - #2) } * .96 }
217         \linewidth \bool_if:NTF #1 { -.8pt } { - 2\ttwd }
218       } { \file_input:n { \l__codedoc_codemo_name_tl } }
219     }
220     \par \smallskip
221   \group_end:
222   \char_set_catcode_ignore:N \%
223 }

```

Variables used in __codedoc_codemo_start: and __codedoc_codemo_end:Nn.

```

224 \iow_new:N    \l__codedoc_codemo_iow
\l__codedoc_codemo_iow 225 \ior_new:N    \l__codedoc_codemo_ior
\l__codedoc_codemo_ior 226 \int_new:N    \l__codedoc_codemo_int
\l__codedoc_codemo_int 227 \tl_const:Nn \l__codedoc_codemo_name_tl {\jobname.codemo.vrb}
\l__codedoc_codemo_name_tl

```

(End of definition for __codedoc_codemo_start: and others.)

\ttwd Dimensions of the width of one typewriter letter under different fontsizes.

Box for saving one typewriter letter temporally.

```

\l__codedoc_ttwd_box 228 \box_new:N \l__codedoc_ttwd_box
229 \clist_map_inline:nn
230 {
231   tiny, scriptsize, footnotesize, small, \@empty,

```

```

232     large, Large,      LARGE,      huge,  Huge
233   }
234   {
235     \dim_new:c { #1 ttwd }
236     \cs_new_protected_nopar:cpn { update#1ttwd }
237     {
238       \hbox_set:Nn \l__codedoc_ttwd_box { \use:c {#1} \ttfamily x }
239       \dim_gset:cn { #1 ttwd } { \box_wd:N \l__codedoc_ttwd_box }
240     }
241     \hook_gput_code:nnn { begindocument / before } { . }
242     { \use:c { update#1ttwd } }
243   }

```

(End of definition for \ttwd and \l__codedoc_ttwd_box. This function is documented on page 2.)

End the optionlist package for l3docstrip.

```

244 </package>

```

Restore the namespace.

```

245 <@@=>

```

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	
\% 192, 222
\, 81, 103
\^ 179
A	
\allowbreak 81, 113
B	
\baselineskip 203, 214
\bfseries 145
bool commands:	
\bool_if:NTF	202, 205, 206, 209, 216, 217
\bool_lazy_or:nnTF 28
\c_false_bool 165
\c_true_bool 170
box commands:	
\box_new:N 71, 228
\box_scale:Nnn 100
\box_use_drop:N 102
\box_wd:N 95, 98, 239
C	
char commands:	
\char_set_catcode_active:N	.. 179
\char_set_catcode_comment:N	.. 192
\char_set_catcode_ignore:N	.. 222
\char_set_catcode_other:N	... 178
clist commands:	
\clist_item:nn 131
\clist_map_inline:nn 229
codedoc internal commands:	
__codedoc_codemo_end:Nn
.....	9, 165, 170, 173, 187
\l__codedoc_codemo_int
.....	194, 196, 203, 214, 224
\l__codedoc_codemo_ior	193, 195, 224
\l__codedoc_codemo_iow
.....	177, 183, 189, 224
\l__codedoc_codemo_name_tl	..
.....	177, 193, 207, 218, 224
\g__codedoc_codemo_ratio_fp	..
.....	3, 166, 171
__codedoc_codemo_start:
.....	9, 163, 168, 173, 173
\g__codedoc_false_tl 154, 157
__codedoc_keys_aux:nnnnn	...
.....	5, 85, 116, 124, 124, 153
\g__codedoc_keys_delimiter_tl
.....	3, 87, 118
__codedoc_keys_emph:n	124, 131, 142
\l__codedoc_keys_emph_int	.. 86, 107
\g__codedoc_keys_font_tl
.....	3, 88, 103, 119
__codedoc_keys_TF_aux:n
.....	7, 148, 149, 150, 151, 151
\l__codedoc_keys_typeset_box
.....	70, 91, 95, 98, 100, 102
\l__codedoc_keys_typeset_tl	..
.....	70, 83, 91, 104
__codedoc_meta_original:n	..
.....	4, 57, 58
__codedoc_meta_patch:n
.....	57, 59, 66, 68
\g__codedoc_true_tl 154, 157
\l__codedoc_ttwd_box	.. 228, 238, 239
\g__codedoc_vbar_tl	.. 113, 155, 157
codemo, codemo* (env.) 2, 8
color commands:	
\color_group_begin: 135
\color_group_end: 137
\color_if_exist:nTF 7
\color_select:n 136, 145
\color_set:nn 8
\color_set:nnn 9
color internal commands:	
\l__color_named_docolour_prop	.. 3

cs commands:	
\cs_if_exist:NTF	55
\cs_new_protected:Npn . . .	173, 187
\cs_new_protected_nopar:Npn .	
.	124, 142, 151, 236
\cs_set:Npn	180
\cs_set_eq:NN	57, 66
\cs_set_protected:Npn	58
\csname	37
D	
\DeclareDocumentCommand	
.	22, 67, 77, 109, 148, 149, 150
\DeclareDocumentEnvironment .	163, 168
\def	30, 31
dim commands:	
\dim_compare:nNnTF	94, 97
\dim_gset:Nn	239
\dim_new:N	235
\dim_set:Nn	199
\do	178
\docextset	1, 22
\dospecials	179
E	
\edef	37
\else	34, 44, 46
\endcsname	37
environments:	
codemo, codemo*	2, 8
keyval	2, 7
exp commands:	
\exp_args:NNo	182
\exp_args:No	136, 145
\expandafter	37
F	
\fbbox	200, 211
\fi	49, 50, 51
file commands:	
\file_input:n	218
\fontfamily	33, 42
fp commands:	
\fp_eval:n	205, 216
G	
group commands:	
\group_begin:	79, 111, 126, 144, 175, 198
\group_end:	105, 122, 140, 146, 191, 221
H	
hbox commands:	
\hbox:n	4, 59, 68
\hbox_set:Nn	91, 238
\hfill	210
hook commands:	
\hook_gput_code:nnn	54, 61, 241
I	
\ifx	43, 45
int commands:	
\int_if_zero:nTF	128
\int_incr:N	196
\int_new:N	226
\int_use:N	183
\int_zero:N	194
ior commands:	
\ior_map_inline:Nn	195
\ior_new:N	225
\ior_open:Nn	193
iow commands:	
\iow_close:N	189
\iow_new:N	224
\iow_newline:	74
\iow_now:Nn	182
\iow_open:Nn	177
J	
\jobname	227
K	
\keys	2, 70
keys commands:	
\keys_define:nn	3, 15, 107
\keys_set_known:nn	24, 25, 80, 82, 112
keyval (env.)	2, 7
L	
\let	36, 38, 41, 178
\linewidth	206, 217

M		T	
<code>\marginparwidth</code>	74, 95, 98	TeX and L ^A T _E X 2 _ε commands:	
<code>\meta</code>	66, 67, 114	<code>\@bsphack</code>	176
msg commands:		<code>\@currentmetafamily</code>	31
<code>\msg_new:nnn</code>	72	<code>\@currenvir</code>	92
<code>\msg_redirect_module:nnn</code>	60	<code>\@elt</code>	38, 40, 41
<code>\msg_warning:nn</code>	99	<code>\@empty</code>	36, 43, 231
N		<code>\@esphack</code>	190
<code>\newlist</code>	161	<code>\@meta@family@list</code>	39
<code>\null</code>	202, 213	<code>\<fontsize>ttwd</code>	2
P		<code>\docextset</code>	1
<code>\par</code>	197, 210, 220	<code>\expand@font@defaults</code>	35
<code>\parbox</code>	203, 214	<code>\f@family</code>	37
<code>\parindent</code>	199	<code>\f@series</code>	45, 48
<code>\ProcessKeyOptions</code>	20	<code>\hspace</code>	2
R		<code>\if@forced@series</code>	32
<code>\relax</code>	41	<code>\keys</code>	2, 3, 6
<code>\RequirePackage</code>	27, 65	<code>\maybe@load@fontshape</code>	47
S		<code>\meta</code>	2, 4
seq commands:		<code>\prepare@family@series@update</code>	4, 30
<code>\seq_gset_item:Nnn</code>	130	<code>\series@maybe@drop@one@m</code>	48
<code>\seq_set_from_clist:Nn</code>	127	<code>\target@series@value</code>	36, 43, 45, 48
<code>\seq_set_map:NNn</code>	133	<code>\textbar</code>	2
<code>\seq_use:Nn</code>	139	<code>\TF</code>	7
<code>\l_tmpa_seq</code>	127, 130, 133	<code>\TFF</code>	7
<code>\l_tmpb_seq</code>	133, 139	<code>\TTF</code>	7
<code>\setcounter</code>	62	<code>\update@series@target@value</code>	38
<code>\setlist</code>	162	<code>\val</code>	2
skip commands:		<code>\verbatim@line</code>	183
<code>\skip_horizontal:n</code>	202, 213	<code>\verbatim@processline</code>	180
<code>\skip_vertical:n</code>	210	<code>\verbatim@start</code>	185
<code>\c_zero_skip</code>	199	<code>\textbar</code>	160
<code>\small</code>	207	<code>\textup</code>	87, 118, 155
<code>\smallskip</code>	197, 220	<code>\TF</code>	2, 148
<code>\space</code>	75	<code>\TFF</code>	148
<code>\string</code>	72, 74, 99	tl commands:	
sys commands:		<code>\tl_const:Nn</code>	157, 158, 159, 227
<code>\sys_if_engine_opentype:TF</code>	160	<code>\tl_if_eq:NnTF</code>	92
<code>\sys_if_engine_ptex_p:</code>	28	<code>\tl_new:N</code>	70
<code>\sys_if_engine_uptex_p:</code>	28	<code>\tl_set:Nn</code>	83
		<code>\TTF</code>	148
		<code>\ttfamily</code>	54, 88, 103, 119, 155, 238
		<code>\ttwd</code>	2, 202, 213, 217, 228

U	V
<code>\upshape</code> 145	<code>\val</code> 2, <u>109</u>
use commands:	
<code>\use:N</code> 238, 242	<code>\verbatiminput</code> 207