

The dashundergaps package*

Frank Mittelbach

June 24, 2018

Abstract

The `dashundergaps` package offers the possibility to replace material in running text with white space in order to build up forms that can be filled in at a later time. By default the gaps are underlined and followed by a gap number in parentheses, but many other designs are possible, e.g., dashes or dots instead of the underline, no gap numbers or a different format for them, gap widening for easier fill-in, etc. There is also a teacher’s mode which shows the normally hidden text in a special (customizable) format.

1 Introduction

The `dashundergaps` package provides a single command `\gap` which takes its argument and produces a gap of a width matching its width. To better mark this gap it is underlined in some form (could be a solid line, a dashed or dotted line or even a wiggling line). Furthermore, each gap might be numbered to be able to easily refer to it. Here is an example in form of a puzzle:

The initial ‘E.’ in Donald E. Knuth’s name stands for _____. The well-known answer to the Ultimate Question _____⁽¹⁾ is 42 according to _____⁽²⁾. The first edition of _____⁽³⁾ celebrates silver anniversary in 2019. Historically speaking, `expl3` stands for _____⁽⁴⁾ even though it is a production language these days.

And here are some hints for it if you want to fill it out:

- | | |
|--|--|
| 1. If everything would be that easy to answer. | 3. Back then known as the doggie book. |
| 2. The author of “Last Chance To See” and a famous radio show. | 4. Old names die hard. manual to get same height |

The answer is given in Section 4 that shows the gaps filled in, which is due to a, so called, teacher mode that can be activated.

As you see some gaps are numbered with a superscript number (not the default setting) while others aren’t. How this is done and how to change the result is explained in the next section.

*A reimplementaion of a package originally written by Luca Merciadri in 2010.

There also exists a “teacher mode” in which the the gaps are filled with the text given in the argument. This can be used to show the correct answers of some test (as we do in Section 4) or to give a sample fill-in for some form, to help people to fill it out correctly. The “teacher mode” will produce the same line breakings because ensures that the fill-ins take the same about of space as the gaps.

Another important feature is the possibility to artificially widen the gaps (compared to the textual material in the argument). After all, when a form is filled by hand people typically need more space to write some text compared to the same text being typeset. So making the gaps simply as wide as the material may result in too little space.

2 The user interface

The `dashundergaps` package is built as a small application on top of the `ulem` package, a package that defines a number of commands for underling $\langle simple-text \rangle$ in various ways.

| | |
|---|---|
| <code>\uline</code> <code>\uwave</code> <code>\dashuline</code> <code>\dotuline</code> | <code>\uline{\langle simple-text \rangle}</code> <code>\uwave{\langle simple-text \rangle}</code> <code>etc.</code> |
|---|---|

This means that by loading `dashundergaps` the `ulem` commands such as `\uline`, `\uwave` and so forth are automatically made available. These commands are used to do most of the work and the current package only makes sure that instead of the words empty boxes of the same width are are used by `ulem`. This way we get underlined gaps of the right size.

By default, `ulem` would change `\emph` to underline text, so for this application it is loaded with the option `normalem` to prevent that from happening.

| | |
|-------------------|--|
| <code>\gap</code> | <code>\gap*[\langle type \rangle]{\langle text \rangle}</code> |
|-------------------|--|

The main command provided by the package is `\gap` which expects a mandatory $\langle text \rangle$ argument containing the material that is used to produce the gap (and normally invisible). By default the gap is underlined (though that could be changed).

The optional $\langle style \rangle$ argument explicitly defines a certain type of underlining: `u` stands for normal underlining (via `\uline`), `w` for a wave line (via `\uwave`), `b` for blank (i.e., no underlining whatsoever), “-” for a dash-line (via `\dashuline`) and finally “.” for underlining with dots (via `\dotuline`).

By default gaps are numbered using the counter `gapnumber` and this number is shown in parentheses after the gap. With the star form the generation of the number can be toggled, i.e., if it is normally produced because of the current option settings it will be suppressed, if it is suppressed through an option it will be typeset. This way one can select the most convenient setting via an option for the whole document and use the `*` to toggle it occasionally.

Since `\gap` uses `ulem` commands it inherits the limitations of these commands, in that only simple text can be used in the $\langle text \rangle$ argument. For example, a `\footnote` couldn’t be used in that place (but then that wouldn’t make much sense in a gap, would it?).

| | |
|------------------------------|--|
| <code>\TeacherModeOn</code> | <code>\TeacherModeOn % show gap material</code> |
| <code>\TeacherModeOff</code> | <code>\TeacherModeOff % stop showing gap material</code> |

There also exists a teacher mode in which the material for the gaps is visible. This can be used to show the expected answers in case `\gap` is used for preparing tests, or to show a sample fill-in of a form. The teacher mode can be turned on or off anywhere in the document using `\TeacherModeOn` or `\TeacherModeOff`, respectively. Alternatively, it can also be set via an option, as we will see below.

| | |
|----------------------------------|--|
| <code>\dashundergapssetup</code> | <code>\dashundergapssetup{<key-value list>}</code> |
|----------------------------------|--|

The package can be loaded with a number of options (discussed in Section 2.1). A probably better approach is to set any options with the declaration `\dashundergapssetup` which is normally used in the preamble, but can in fact be used throughout the document to change settings on the fly. It only changes options explicitly given so can be used to overwrite some defaults but leave everything else unchanged.

2.1 Options to customize the gap display

All of the package options are implemented as key/value options. In case of boolean options one can just give the option name as a short for setting the option to `true`. Most options can be used during package loading in the optional argument of `\usepackage`. However if the value requires some L^AT_EX code (e.g., `gap-font` that expects a font declaration command) then this will not work due to some limitations in the current L^AT_EX package loader. For such options use `\dashundergapssetup` instead which will always work.

2.1.1 Gap modus

The general processing mode is defined through the following options:

teacher-mode Boolean that turns on teacher mode (i.e., the gap material will be come visible if set to `true`). It's default is `false`.

gap-mode Boolean that is the inverse of **teacher-mode** and just provided for convenience, i.e., you can use it as a short form for **teacher-mode=false**.

teachermode Alternative name for **teacher-mode** because that is what it was called in the first package release.

2.1.2 Gap formatting

Formatting of the gaps is handled by the following five options:

gap-format A choice option that defines how the gap is marked. It accepts the following values: **underline** (default), **dash**, **dot**, **wave** or **blank**.

teacher-gap-format Another choice option to be used in case we are in “teacher mode”. It accepts the same values, but this time the default is **blank** as normally the gap text is typeset in bold font and is therefore already identifiable and doesn't additionally underlining or some sort. However, depending on the circumstances it might be helpful to keep the underlining (or use a different kind of underlining) while in “teacher mode”.

gap-font This option expects a font directive as its value, e.g., `\bfseries` (which is also the default). Using this option without supplying a value is equivalent to supplying an empty value. It will be used to determine the font for the gap material regardless of the mode. This is important to ensure that the gaps always have the same width regardless of whether or not the material is shown.

For the example puzzle above it was set to `\itshape` which you can see in the puzzle answer.

dash Short name for `gap-format=dash`.

dot Short name for `gap-format=dot`.

2.1.3 Gap numbers

Producing the gap numbers is handled by the following options:

gap-numbers Boolean that determines whether or not gap numbers are displayed. Default is `true`.

gap-number-format Code that is executed if gap numbers are produced. Default is `\textnormal{_(\thegapnumber)}`.

numbers Short name for `gap-numbers`.

There is also a way to control displaying the total number of gaps:

display-total-gaps Boolean to determine if the total number of gaps should be shown at the very end of the document. Default is `false`.

displaynbgaps This is just another name for the same boolean that was used in the first version of the package.

2.1.4 Gap widening

Finally, for extending the gap width we have these options:

gap-widen Boolean that decides if the gaps should be made wider or not (default is `false` but mainly for historical reasons).

gap-extend-minimum Minimum of extra space that should be added to each gap if gap widening is active. Default is `10pt`.

gap-extend-percent Percentage (as a number) by which the gap should be made wider if widening is active. The result is compared to `gap-extend-minimum` and the larger of the two is used. Default is `20`.

widen Short name for `gap-widen`.

3 Differences to the original package

The main user interface of the two versions is identical so that it is possible to use the new version as a drop-in replacement for the old. However, the feature set in form of key/value options has been greatly extended offering functionality that have been previously unavailable. Furthermore a number of bugs have been corrected (and possibly new ones introduced).

- Stray spaces in the definition of `\gap` (that showed up in the output) have been eliminated;
- Various combinations of options that didn't work are now possible;
- Nesting isn't possible for obvious reasons, but the fact is now detected and catered for by ignoring the inner gap requests after generating an error;
- Option names have been normalized (though the original names are still available);
- The option `phantomtext` is no longer necessary, though still supported (with a warning) as a no-op;
- The names of the \LaTeX counters used have changed, so if you directly addressed them that would need changing;
- The font used in teacher mode (by default boldface) is now also being used if gap mode is chosen, to ensure that the output in all modes will produce identical line breaks; For that reason the `pkgulem` machinery is always used, even if not underlining (or dashing, etc.) is needed.
- The gaps can be extended by a percentage or by some minimal amount to ensure that there is enough space to fill in the text (given that hand-written text is typically wider than typeset material. The values are adjustable;
- `\gap` has now an optional argument through which you can explicitly request the type on underlining you want to use;
- It also supports a star form which toggles the setting of gap numbers;
- The Use of `\label` within the `\gap` command argument allows to later refer to that gap by its number (provided a gap number is typeset);
- The implementation is done with `expl3` the programming language for $\text{\LaTeX}3$.

4 Solution to the puzzle

Here we repeat the puzzle from above with `\TeacherModeOn`.

The initial 'E.' in Donald E. Knuth's name stands for *Erwin*. The well-known answer to the Ultimate Question *of Life, the Universe, and Everything* ⁽⁵⁾ is 42 according to *Douglas Adams* ⁽⁶⁾. The first edition of *The \LaTeX Companion* ⁽⁷⁾ celebrates silver anniversary in 2019. Historically speaking, `expl3` stands for *EXperimental Programming Language 3* ⁽⁸⁾ even though it is a production language these days.

This was produced using the following changes to the defaults:

```
\dashundergapssetup{
  ,gap-number-format = \,\textsuperscript{\normalfont(\thegapnumber)}
  ,gap-font           = \itshape
  ,teacher-gap-format = underline
  ,gap-widen
}
```

As you can see we use `\itshape` for the font (to be able to show the bold face in one of the answers) and also forced underlining in teacher mode to better show the gap widening. The gap number is raised and we separate it a tiny bit from the gap material. We also use `\normalfont` in the formatting to ensure that the gap number is set upright and not in italic shape.

5 The Implementation

5.1 Loading and fixing ulem

First thing to do is to load ulem without changing `\emph` or `\em`:

```
1 \package
2 \RequirePackage[normalem]{ulem}
```

The code in this section follows L^AT_EX 2_ε conventions, i.e., models the commands like they look in ulem itself.

\dotuline The dot produce by `\dotuline` depends on the current font which is a somewhat questionable design (if you underline a part with a single bold word somewhere inside it will change the shape of the dot line). So we always use the `\normalfont` dot.

```
3 \def\dotuline{\bgroup
4   \ULsetULdepth
5   \markoverwith{\begingroup
6     \advance\ULdepth0.08ex
7     \lower\ULdepth\hbox{\normalfont \kern.1em .\kern.04em}%
8     \endgroup}%
9   \ULon}
```

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

\@ublack `\@ublack` underlines with blanks. Normally not really useful (which is why we make it internal), but when we want to have ulem acting—but without actually underlining with something visible—that is the command to use.

```
10 \def\@ublack{\bgroup\let\UL@leadtype\@empty\ULon}
```

(End definition for `\@ublack`.)

\phantom@putbox This is going to be a replacement for `\UL@putbox` when we want ulem to underline something but show a gap instead of the material.

```
11 \def\phantom@putbox{\ifx\UL@start\@empty \else % not inner
12   \vrule\@width\z@ \LA@penalty\@M
13   {\UL@skip\wd\UL@box \UL@leaders \kern-\UL@skip}%
14   \hb@xt@\wd\UL@box{}%
15   \fi}
```

(End definition for `\phantom@putbox`.)

5.2 The main implementation part

The rest of the package is written in `expl3`. We use `fmdug` as our internal prefix.

```

16 <@@=fmdug>
    We need xparse and l3keys2e
17 \RequirePackage{xparse,l3keys2e}
18 \ProvidesExplPackage
19   {dashundergaps}
20   {2018/06/24}
21   {v2.0a}
22   {Dashing and underlining (phantom) text}

```

5.2.1 User interface commands

`\gap` Gap takes a star, optional and mandatory argument and calls `__fmdug_gap:nnn` to do the work.

```

23 \DeclareDocumentCommand \gap { som } { \__fmdug_gap:nnn {#1}{#2}{#3} }

```

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

`\dashundergapssetup` Change options anywhere.

```

24 \NewDocumentCommand \dashundergapssetup { m }
25   { \keys_set:nn {fmdug} {#1} \ignorespaces }

```

(End definition for `\dashundergapssetup`. This function is documented on page 3.)

`\TeacherModeOn` We provide shortcuts for turning teacher mode on or off.

`\TeacherModeOff`

```

26 \DeclareDocumentCommand \TeacherModeOn {}
27   { \bool_set_true:N \l__fmdug_teacher_bool }
28 \DeclareDocumentCommand \TeacherModeOff {}
29   { \bool_set_true:N \l__fmdug_teacher_bool }

```

(End definition for `\TeacherModeOn` and `\TeacherModeOff`. These functions are documented on page 3.)

5.2.2 Counters

`\c@gapnumber` We have one user-level counter which is referenceable and holds the gap number of the current gap. It can be reset to 0 to restart counting.

```

30 \newcounter{gapnumber}

```

(End definition for `\c@gapnumber`.)

`\c@totalgapnumber` We also keep track of all gaps ever made using another user-level counter.

```

31 \newcounter{totalgapnumber}

```

(End definition for `\c@totalgapnumber`.)

`\l__fmdug_extend_dim` A help register to calculate the gap width later on.

```

32 \dim_new:N \l__fmdug_extend_dim

```

(End definition for `\l__fmdug_extend_dim`.)

`\l__fmdug_extra_left_gap_tl` Two scratch token lists to enlarge the gap on the left or right side.

`\l__fmdug_extra_right_gap_tl`

```

33 \tl_new:N \l__fmdug_extra_left_gap_tl
34 \tl_new:N \l__fmdug_extra_right_gap_tl

```

(End definition for `\l__fmdug_extra_left_gap_tl` and `\l__fmdug_extra_right_gap_tl`.)

5.2.3 Messages

```

35 \msg_new:nnn {dashundergaps} {deprecated}
36 {
37   The~ #1~ ‘#2’~ you~ used~ \msg_line_context: \ is~ deprecated~and~ there~
38   is~ no~ replacement.~ Since~ I~ will~ not~ guarantee~ that~ #1~ ‘#2’~
39   will~ be~ kept~ forever~ I~ strongly~ encourage~ you~ to~ remove~ it~
40   from~ your~ document.
41 }
42 \msg_new:nnnn {dashundergaps} {nested}
43 { The~ \gap command~ can’t~ be~ tested! }
44 { Nesting~ doesn’t~ make~ much~ sense~ as~ the~ inner~ one~ wouldn’t~ be~
45   visible.~ ~ To~ allow~ continuation~ it~ is~ handled~ as~ if~ it~ wasn’t~
46   been~ asked~ for. }

```

5.2.4 Option handling

Here we define all option keys for use as package options or inside `\dashundergapssetup`. these are all straight forward assignments to variables. These internal variables are declared by the key declarations if unknown, so they are not separately declared before.

```

47 \keys_define:nn {fmdug}
48 {
49   % =====
50   ,teacher-mode      .bool_set:N = \l__fmdug_teacher_bool
51   ,teacher-mode      .default:n  = true
52   ,teacher-mode      .initial:n  = false
53   % -----
54   ,gap-mode          .bool_set_inverse:N = \l__fmdug_teacher_bool
55   % =====
56   ,gap-format
57     .choice:
58   ,gap-format / underline
59     .code:n = \tl_set:Nn \l__fmdug_gap_format_tl { \uline }
60   ,gap-format / dash
61     .code:n = \tl_set:Nn \l__fmdug_gap_format_tl { \dashuline }
62   ,gap-format / dot
63     .code:n = \tl_set:Nn \l__fmdug_gap_format_tl { \dotuline }
64   ,gap-format / wave
65     .code:n = \tl_set:Nn \l__fmdug_gap_format_tl { \uwave }
66   ,gap-format / blank
67     .code:n = \tl_set:Nn \l__fmdug_gap_format_tl { \@ublack }
68   ,gap-format / unknown
69     .code:n = \tl_set:Nn \l__fmdug_gap_format_tl { #1 }
70   ,gap-format
71     .initial:n = underline
72   % =====
73   ,teacher-gap-format
74     .choice:
75   ,teacher-gap-format / underline
76     .code:n = \tl_set:Nn \l__fmdug_teacher_gap_format_tl { \uline }
77   ,teacher-gap-format / dash
78     .code:n = \tl_set:Nn \l__fmdug_teacher_gap_format_tl { \dashuline }
79   ,teacher-gap-format / dot
80     .code:n = \tl_set:Nn \l__fmdug_teacher_gap_format_tl { \dotuline }

```



```

81 ,teacher-gap-format / wave
82   .code:n = \tl_set:Nn \l__fmdug_teacher_gap_format_tl { \uwave }
83 ,teacher-gap-format / blank
84   .code:n = \tl_set:Nn \l__fmdug_teacher_gap_format_tl { \@ublack }
85 ,teacher-gap-format / unknown
86   .code:n = \tl_set:Nn \l__fmdug_teacher_gap_format_tl { #1 }
87 ,teacher-gap-format
88   .initial:n = blank
89   % =====
90 ,gap-widen      .bool_set:N = \l__fmdug_gap_widen_bool
91 ,gap-widen      .default:n = true
92 ,gap-widen      .initial:n = false
93   % -----
94 ,widen          .meta:n = { gap-widen }
95   % -----
96 ,gap-extend-minimum .dim_set:N = \l__fmdug_gap_min_dim
97 ,gap-extend-minimum .initial:n = 10pt
98   % -----
99 ,gap-extend-percent .tl_set:N = \l__fmdug_gap_percent_tl
100 ,gap-extend-percent .initial:n = 20
101   % =====
102 ,gap-numbers     .bool_set:N = \l__fmdug_number_bool
103 ,gap-numbers     .default:n = true
104 ,gap-numbers     .initial:n = true
105   % -----
106 ,numbers        .meta:n = { gap-numbers }
107   % -----
108 ,gap-number-format .tl_set:N = \l__fmdug_gapnum_format_tl
109 ,gap-number-format .initial:n = \textnormal{\space (\thegapnumber)}
110   % =====
111 ,display-total-gaps .bool_gset:N = \g__fmdug_display_total_gaps_bool
112 ,display-total-gaps .default:n = true
113 ,display-total-gaps .initial:n = false
114   % =====
115 ,gap-font        .tl_set:N = \l__fmdug_font_tl
116 ,gap-font        .default:n =
117 ,gap-font        .initial:n = \bfseries

```

And finally the original options:

```

118   % =====
119 ,teachermode     .meta:n = { teacher-mode }
120 ,dash           .meta:n = { gap-format = dash }
121 ,dot            .meta:n = { gap-format = dot }
122 ,displaynbgaps .meta:n = { display-total-gaps }
123   % -----
124 ,phantomtext
125   .code:n = \msg_warning:nnnn{dashundergaps}{deprecated}
126               {option}{phantomtext}
127   % =====
128 }

```

`__fmdug_gap:nnn` And here finally comes the action. `__fmdug_gap:nn` expects two arguments: `#1` indicates what kind of “underlining” is wanted (anything not recognized is ignored, in particular

“–NoValue–” if `\gap` was used without optional argument) and #2 is the material to produce a gap for.

```
129 \cs_new:Npn \__fmdug_gap:nnn #1#2#3 {
130   \group_begin:
```

Define the font used inside the gap. We need to do this up front since we want to measure the text (and that needs the correct font already).

```
131   \l__fmdug_font_tl
```

Nesting is not supported so inside the gap we redefine `__fmdug_gap:nnn` to raise an error and just return the third argument if it is encountered again.

```
132   \cs_set:Npn \__fmdug_gap:nnn ##1##2##3
133   {
134     \msg_error:nn{dashundergaps}{nested}
135     ##3
136   }
```

We increment the counter for the total number of gaps always, but the `gapnumber` only if we are displaying it. For the latter one we use `\refstepcounter` to make it referenceable.

```
137   \stepcounter{totalgapnumber}
138   \bool_xor:nnT { #1 } { \l__fmdug_number_bool }
139   { \refstepcounter{gapnumber} }
```

Next we prepare for widening if that is being asked for: Measure the width of the text and then set `\l__fmdug_extend_dim` to be the requested percentage divided by two of that width (since we add it later on both sides).

Then compare it to the minimum / 2 and choose whatever is larger.

```
140   \bool_if:NTF \l__fmdug_gap_widen_bool
141   {
142     \settoheight \l__fmdug_extend_dim {#3}
143     \dim_set:Nn \l__fmdug_extend_dim
144       { \l__fmdug_gap_percent_tl \l__fmdug_extend_dim / 200 }
145   %
146     \dim_compare:nNnT \l__fmdug_extend_dim < { .5\l__fmdug_gap_min_dim }
147     { \dim_set:Nn \l__fmdug_extend_dim { .5\l__fmdug_gap_min_dim } }
```

Now we prepare what needs to go to the left and the right of the gap.

```
148   \tl_set:Nn \l__fmdug_extra_left_gap_tl
149     { \hbox_to_wd:nn\l__fmdug_extend_dim{} \allowbreak }
150   \tl_set:Nn \l__fmdug_extra_right_gap_tl
151     { \allowbreak \hbox_to_wd:nn\l__fmdug_extend_dim{} }
152   }
```

And if no widening is asked for we clear these two token lists so that they don't do anything.

```
153   {
154     \tl_clear:N \l__fmdug_extra_left_gap_tl
155     \tl_clear:N \l__fmdug_extra_right_gap_tl
156   }
```

Next comes deciding the gap format. If in teacher mode it will be whatever is in `\l__fmdug_teacher_gap_tl`. Otherwise it is either based on the content of the optional argument or if that is not given or unknown it will be `\l__fmdug_gap_format_tl`.

```
157   \bool_if:NTF \l__fmdug_teacher_bool
158   { \l__fmdug_teacher_gap_format_tl }
```

```

159 { \cs_set_eq:NN \UL@putbox \phantom@putbox
160   \str_case:nnF {#2}
161   {
162     {u} { \uline }
163     {w} { \uwave }
164     {b} { \@ublack }
165     {.} { \dotuline }
166     {-} { \dashuline }
167   }
168   { \l__fmdug_gap_format_tl }
169 }

```

Whatever was decided as gap format it needs one argument, i.e., the material (with possible gap extension on both sides.

```

170 { \l__fmdug_extra_left_gap_tl #3 \l__fmdug_extra_right_gap_tl }

```

Finally we typeset the gap number is that was requested.

```

171 \bool_xor:nnT { #1 } { \l__fmdug_number_bool }
172 { \l__fmdug_gapnum_format_tl }
173 \group_end:
174 }

```

(End definition for __fmdug_gap:nnn.)

`__fmdug_display_total_gaps:` This command will display the total number of gaps if requested. The hard-wired formatting comes from the first version of the package.

```

175 \cs_new:Npn \__fmdug_display_total_gaps: {
176   \vfill \centering
177   \bfseries Total~ Gaps:~ \thetotalgapnumber
178 }

```

(End definition for __fmdug_display_total_gaps:.)

5.2.5 Closing shop

At the end of the document we typeset the total number of gaps if requested.

```

179 \AtEndDocument{
180   \bool_if:NT \g__fmdug_display_total_gaps_bool
181     \__fmdug_display_total_gaps:
182 }

```

So what remains to be done is execute all options passed to the package via `\usepackage`.

```

183 \ProcessKeysPackageOptions{fmdug}
184 \*package

```

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.

| | |
|---|--|
| A | |
| <code>\allowbreak</code> | 149, 151 |
| <code>\AtEndDocument</code> | 179 |
| B | |
| <code>\bfseries</code> | 117, 177 |
| bool commands: | |
| <code>\bool_if:NTF</code> | 140, 157, 180 |
| <code>\bool_set_true:N</code> | 27, 29 |
| <code>\bool_xor:nnTF</code> | 138, 171 |
| C | |
| <code>\centering</code> | 176 |
| cs commands: | |
| <code>\cs_new:Npn</code> | 129, 175 |
| <code>\cs_set:Npn</code> | 132 |
| <code>\cs_set_eq:NN</code> | 159 |
| D | |
| <code>\dashuline</code> | 2, 2, 61, 78, 166 |
| <code>\dashundergapssetup</code> | 3, 8, 24 |
| <code>\DeclareDocumentCommand</code> | 23, 26, 28 |
| dim commands: | |
| <code>\dim_compare:nNnTF</code> | 146 |
| <code>\dim_new:N</code> | 32 |
| <code>\dim_set:Nn</code> | 143, 147 |
| <code>\dotuline</code> | 2, 2, 3, 6, 63, 80, 165 |
| E | |
| <code>\em</code> | 6 |
| <code>\emph</code> | 2, 6 |
| F | |
| fmdug internal commands: | |
| <code>__fmdug_display_total_gaps:</code> <u>175</u> , 181 | |
| <code>\g__fmdug_display_total_gaps_</code> <code>bool</code> | 111, 180 |
| <code>\l__fmdug_extend_dim</code> | 10, 32, 142, 143, 144, 146, 147, 149, 151 |
| <code>\l__fmdug_extra_left_gap_tl</code> | 33, 148, 154, 170 |
| <code>\l__fmdug_extra_right_gap_tl</code> | 33, 150, 155, 170 |
| <code>\l__fmdug_font_tl</code> | 115, 131 |
| <code>__fmdug_gap:nn</code> | 9 |
| <code>__fmdug_gap:nnn</code> | 7, 10, 23, <u>129</u> |
| <code>\l__fmdug_gap_format_tl</code> | 10, 59, 61, 63, 65, 67, 69, 168 |
| <code>\l__fmdug_gap_min_dim</code> | 96, 146, 147 |
| <code>\l__fmdug_gap_percent_tl</code> | 99, 144 |
| <code>\l__fmdug_gap_widen_bool</code> | 90, 140 |
| <code>\l__fmdug_gapnum_format_tl</code> | 108, 172 |
| <code>\l__fmdug_number_bool</code> | 102, 138, 171 |
| <code>\l__fmdug_teacher_bool</code> | 27, 29, 50, 54, 157 |
| <code>\l__fmdug_teacher_gap_format_tl</code> | 76, 78, 80, 82, 84, 86, 158 |
| <code>\l__fmdug_teacher_gap_tl</code> | 10 |
| <code>\footnote</code> | 2 |
| G | |
| <code>\gap</code> | 1, 2, 3, 5, 10, <u>23</u> , 43 |
| group commands: | |
| <code>\group_begin:</code> | 130 |
| <code>\group_end:</code> | 173 |
| H | |
| hbox commands: | |
| <code>\hbox_to_wd:nn</code> | 149, 151 |
| I | |
| <code>\ignorespaces</code> | 25 |
| K | |
| keys commands: | |
| <code>\keys_define:nn</code> | 47 |
| <code>\keys_set:nn</code> | 25 |
| L | |
| <code>\label</code> | 5 |
| M | |
| msg commands: | |
| <code>\msg_error:nn</code> | 134 |
| <code>\msg_line_context:</code> | 37 |
| <code>\msg_new:nnn</code> | 35 |
| <code>\msg_new:nnnn</code> | 42 |
| <code>\msg_warning:nnnn</code> | 125 |
| N | |
| <code>\newcounter</code> | 30, 31 |
| <code>\NewDocumentCommand</code> | 24 |
| <code>\normalfont</code> | 6, 6, 7 |
| P | |
| <code>\ProcessKeysPackageOptions</code> | 183 |
| <code>\ProvidesExplPackage</code> | 18 |

| | | | |
|--|--------------------|---------------------------------|---|
| R | | U | |
| <code>\refstepcounter</code> | 10, 139 | <code>\phantom@putbox</code> | 11, 159 |
| <code>\RequirePackage</code> | 2, 17 | <code>\UL@putbox</code> | 6 |
| S | | <code>\textnormal</code> | 109 |
| <code>\settowidth</code> | 142 | <code>\thegapnumber</code> | 109 |
| <code>\space</code> | 109 | <code>\thetotalgapnumber</code> | 177 |
| <code>\stepcounter</code> | 137 | tl commands: | |
| str commands: | | <code>\tl_clear:N</code> | 154, 155 |
| <code>\str_case:nnTF</code> | 160 | <code>\tl_new:N</code> | 33, 34 |
| | | <code>\tl_set:Nn</code> | 59, 61, 63, 65, 67, 69, 76, 78, 80, 82, 84, 86, 148, 150 |
| T | | V | |
| <code>\TeacherModeOff</code> | 3, 26 | <code>\uline</code> | 2, 2, 59, 76, 162 |
| <code>\TeacherModeOn</code> | 3, 5, 26 | <code>\usepackage</code> | 3, 11 |
| TeX and L ^A T _E X 2 _ε commands: | | <code>\uwave</code> | 2, 2, 65, 82, 163 |
| <code>\@ublack</code> | 6, 10, 67, 84, 164 | | |
| <code>\c@gapnumber</code> | 30 | | |
| <code>\c@totalgapnumber</code> | 31 | | |