

enumext

ENUMERATE EXERCISE SHEETS

V1.0 2024-11-03^{*}

©2024 by Pablo González[†]

CTAN: <https://www.ctan.org/pkg/enumext>

 <https://github.com/pablgonz/enumext>

Abstract

This package provides enumerated list environments compatible with *tagging* PDF for creating “*simple exercise sheets*” along with “*multiple choice questions*”, storing the “*answers*” to these in memory using `multicol` and `scontents` packages.

Contents

| | | | | | |
|----------|---|----------|-----------|--|------------|
| 1 | Introduction | 1 | 6 | The storage system | 11 |
| 1.1 | Description and usage | 2 | 6.1 | Keys for storage system | 11 |
| 1.2 | The concept of left margin | 3 | 6.1.1 | Keys for label and ref | 12 |
| 1.3 | User interface | 3 | 6.1.2 | Keys for wrap and display | 12 |
| 1.3.1 | Internal counters | 3 | 6.1.3 | Keys for debug and checking | 12 |
| 1.3.2 | Public dimension | 3 | 6.2 | The command <code>\anskey</code> | 13 |
| 1.3.3 | Support for <code>multicol</code> | 4 | 6.2.1 | Keys for <code>\anskey</code> | 13 |
| 1.3.4 | Support for <code>minipage</code> | 4 | 6.3 | The environment <code>anskey*</code> | 13 |
| 1.3.5 | The <code>\label</code> and <code>\ref</code> system | 4 | 6.3.1 | Keys for <code>anskey*</code> | 14 |
| 1.3.6 | Support for <code>\footnote</code> | 4 | 6.4 | The environment <code>keyans</code> | 14 |
| 2 | The environments provided | 5 | 6.4.1 | The <code>\item*</code> in <code>keyans</code> | 15 |
| 2.1 | The environment <code>enumext</code> | 5 | 6.5 | The environment <code>keyanspic</code> | 15 |
| 2.2 | The environment <code>enumext*</code> | 5 | 6.5.1 | Keys for <code>keyanspic</code> | 16 |
| 2.3 | The command <code>\item*</code> | 5 | 6.5.2 | The command <code>\anspic</code> | 16 |
| 2.3.1 | Keys for <code>\item*</code> | 6 | 6.6 | Printing stored content | 17 |
| 2.4 | The command <code>\item</code> in <code>enumext*</code> | 6 | 6.6.1 | The command <code>\getkeyans</code> | 17 |
| 3 | The command <code>\setenumext</code> | 6 | 6.6.2 | The command <code>\foreachkeyans</code> | 17 |
| 4 | The command <code>\setenumextmeta</code> | 6 | 6.6.3 | The command <code>\printkeyans</code> | 18 |
| 5 | The <code>keyval</code> system | 7 | 7 | Full examples | 19 |
| 5.1 | Keys for label and ref | 7 | 8 | Tagged PDF examples | 21 |
| 5.2 | Keys for spaces | 8 | 9 | The way of non-enumerated lists | 22 |
| 5.2.1 | Vertical spaces | 8 | 10 | References | 24 |
| 5.2.2 | Horizontal spaces | 9 | 11 | Change history | 24 |
| 5.3 | Keys for add code | 9 | 12 | Index of Documentation | 25 |
| 5.4 | Keys for start, series and resume | 10 | 13 | Implementation | 27 |
| 5.5 | Keys for <code>multicols</code> | 10 | 14 | Index of Implementation | 143 |
| 5.6 | Keys for <code>minipage</code> | 11 | | | |
| 5.6.1 | The command <code>\miniright</code> | 11 | | | |
| 5.6.2 | The key <code>mini-right</code> | 11 | | | |

Motivation and acknowledgments

Usually it is enough to use the classic `enumerate` environment to generate “*simple exercise sheets*” or “*multiple choice questions*”, the basic idea behind `enumext` is to cover three points:

1. To have a simple interface to be able to write “*lists of exercises*” with “*answers*”.
2. To have a simple interface for writing “*multiple choice questions*”.
3. To have a simple interface for placing “*columns*” and “*drawings*” or “*tables*”.

This package would not be possible without Phelype Oleinik who has collaborated and adapted a large part of the code and all \LaTeX team for their great work and to the different members of the \TeX-SX community who have provided great answers and ideas. Here a note of the main ones:

1. Answer given by Alan Munn in `\topsep`, `\itemsep`, `\partopsep`, `\parsep` - what do they each mean (and what about the bottom)?
2. Answer given by Enrico Gregorio in `Understanding minipages` - aligning at top
3. Answer given by Ulrich Diez in `Different mechanics of hyperlink vs. hyperref`
4. Answer given by Enrico Gregorio in `Minipage and multicols`, vertical alignment

^{*}This file describes a documentation for v1.0, last revised 2024-11-03.

[†]E-mail: pablgonz@educarchile.cl.

License and Requirements

Permission is granted to copy, distribute and/or modify this software under the terms of the LaTeX Project Public License (lpp), version 1.3 or later (<https://www.latex-project.org/lpp1.txt>). The software has the status “maintained”.
The enumext package loads and requires multicol[3] and contents[4] packages, need to have a modern TeX distribution such as TeX Live or MiKTeX. It has been tested with the standard classes provided by L^AT_EX: book, report, article and letter on 10pt, 11pt and 12pt.

The minimum requirement is L^AT_EX release 2024-11-01.

1 Introduction

In the L^AT_EX world there are many useful packages and classes for creating “lists of exercises”, “worksheets” or “multiple choice questions”, classes like exam[1] and packages like xsim[2] do the job perfectly, but they don’t always fit the basic day to day needs.

In my work (and in the work of many teachers) it is common to use “simple exercise sheets” also known as “informal lists of exercises”, as an example:

1. Factor $x^2 - 2x + 1$

2. Factor $3x + 3y + 3z$

3. True False

(a) $\alpha > \delta$

(b) L^AT_EXze is cool?

4. Related to Linux
- (a) You use linux?

(b) Usually uses the package manager?

(c) Rate the following package and class

i. xsim-exam

ii. xsim

iii. exsheets

Sometimes we are also interested in showing the “answers” along with the questions:

1. Factor $x^2 - 2x + 1$

*

$(x - 1)^2$

2. Factor $3x + 3y + 3z$

*

$3(x + y + z)$

3. True False

(a) $\alpha > \delta$

*

False

(b) L^AT_EXze is cool?

*

Very True!

4. Related to Linux
- (a) You use linux?

*

Yes

(b) Usually uses the package manager?

*

Yes, dnf

(c) Rate the following package and class

i. xsim-exam

*

doesn’t exist for now :(

ii. xsim

*

very good

iii. exsheets

*

obsolete
- Or we are interested in referring to a specific question and its “answer”, for example:
- The answer to 3.(b) is “Very True!” and the answer to 4.(c).ii is “very good”.
- Or we are interested in printing all the “answers”:
1. $(x - 1)^2$

2. $3(x + y + z)$

3. (a) False

(b) Very True!

4. (a) Yes

⌘

(b) Yes, dnf

⌘

(c) i. doesn’t exist for now :(

⌘

ii. very good

⌘

iii. obsolete

⌘
- Another very common thing to use in my work is “multiple choice questions”, for example:
1. First type of questions

A) value

B) correct

C) value

D) value

2. Second type of questions

I. $2\alpha + 2\delta = 90^\circ$

II. $\alpha = \delta$

III. $\angle EDF = 45^\circ$

A) I only

B) II only

C) I and II only

D) I and III only

E) I, II, and III

* 3. Third type of questions

(1) $2\alpha + 2\delta = 90^\circ$

(2) $\angle EDF = 45^\circ$

A) value

B) value

C) value

D) value

E) value

4. Question with image and label below:

A

A)

B


B)

A

C)

A

D)



E)

5. Question with image on right side:

A) value

B) value

C) value

D) correct

E) value

B

©2024 by Pablo González L

2 / 158

Where what we are interested in the $\langle label \rangle$ and a “short note” that we leave as an explanation, and then print them:

1. B) $x = 5$

2. D)

3. C) some note
- ⌘ 4. E) A duck

⌘ 5. D) “other note”

⌘

The `enumext` package was created and designed to meet these small requirements in the creation of “simple worksheets” and “multiple choice questions”.

- These “simple worksheets” or “multiple choice questions” appear to be easy to obtain using a combination of the `enumerate`, `minipage` and `multicols` environments, but like many things, what “looks simple” is not so simple.

1.1 Description and usage

The `enumext` package defines enumerated environments using the `list` environment provided by \TeX , but “does not redefine” any internal commands associated with it such as `\list`, `\endlist` or `\item` outside of the “scope” in which they are defined.

- This package is NOT intend to replace the `enumerate` environment nor replace the powerful `enumitem`[6], the approach is intended to work without hindering either of them.

This package can be used with `xelatex`, `lualatex`, `pdflatex` and the classical `latex`»`dvips`»`ps2pdf` and is present in \TeX Live and $\text{MiK}\text{\TeX}$, use the package manager to install. For manual installation, download `enumext.zip` and unzip it, run `luatex enumext.ins` and move all files to appropriate locations, then run `mktexlsr`. To produce the documentation run `arara enumext.dtx`.

| | | |
|--------------------------|---|--|
| <code>enumext.sty</code> | » | <code>TDS:tex/latex/enumext/</code> |
| <code>enumext.pdf</code> | » | <code>TDS:doc/latex/enumext/</code> |
| <code>README.md</code> | » | <code>TDS:doc/latex/enumext/</code> |
| <code>enumext.dtx</code> | » | <code>TDS:source/latex/enumext/</code> |
| <code>enumext.ins</code> | » | <code>TDS:source/latex/enumext/</code> |

The package is loaded in the usual way:

```
\usepackage{enumext}
```

1.2 The concept of left margin

There is a direct relationship between the parameters `\leftmargin`, `\itemindent`, `\labelwidth` and `\labelsep` plus an “extra space” that makes it difficult to obtain the desired *horizontal spaces* in a `list` environment. Usually we don’t want the `list` to go beyond the left margin of the page, but since these four values are related, that causes a problem.

The `enumitem`[6] package adds the `\labelindent` parameter to solve some of these problems. A simplified representation of this in the figure 1.



Figure 1: Representation of horizontal lengths in `enumitem`.

The `enumext` package does NOT provide a user interface to set the values for `\leftmargin` and `\itemindent`, instead it provides the keys `list-offset` and `list-indent` which internally set the values for `\leftmargin` and `\itemindent`. The concepts of `\leftmargin` and `\itemindent` are different in `enumext`. The figure 2 shows the visual representation of idea.



Figure 2: Representation of horizontal lengths concept in `enumext`.

In this way we reduce a *little* the amount of parameters we have to pass. With the default values of keys `list-offset`, `list-indent`, `labelwidth` and `labelsep` the lists will have the (usually) expected output for “simple worksheets”. The figure 3 shows the visual representation.



Figure 3: Default horizontal lengths `list-offset=0pt`, `list-indent=\labelwidth+\labelsep` in `enumext`.

1.3 User interface

The user interface consists of two main list environments `enumext` (vertical) and `enumext*` (horizontal), the environment `anskey*` and the command `\anskey` to “store content” and the environments `keyans`, `keyans*` and `keyanspic` for multiple choice. It also provides the commands `\getkeyans` to print individual *stored content*, `\printkeyans` and `\foreachkeyans` to print all *stored content*, `\miniright` for `minipage`, `\setenumext` and `\setenumextmeta` to config [*key* = *val*] options.

1.3.1 Internal counters

The package `enumext` uses internally the `enumXi`, `enumXii`, `enumXiii`, `enumXiv` counters for the four nesting levels of the `enumext` environment, the `enumXv` counter for the `keyans` environment, the `enumXvi` counter for the `keyanspic` environment, the counter `enumXvii` for `enumext*` environment and the counter `enumXviii` for `keyans*` environment.

- If any package defines these counters or they are user-defined in the document, the package will return a fatal error and abort the load.

1.3.2 Public dimension

The package `enumext` only provides a single public dimension `\itemwidth` and is intended for user convenience only and is not for internal use as such. The dimension `\itemwidth` is *rigid length* and contains the “width of the content” of each `\item` regardless of `labelwidth` and `labelsep`.

- If any package defines `\itemwidth` or they are user-defined `\itemwidth` in the document, the package will overwrite it without warning.

1.3.3 Support for multicol

The package provides direct support for using the `multicol`[3] package. This allows to obtain directly a two-column output as shown in the figure 4.



Figure 4: Representation of the two column output for a nested level in `enumext` environment.

The “non starred” version of the `multicols` environment is always used together with the `\raggedcolumns` command and is controlled by `columns` and `columns-sep` keys. It can be used in all nesting levels of the environment `enumext` and the environment `keyans` and can together with the `mini-env` key. If you need to force a start a new column `\columnbreak` must be used (see §5.5).

- The `\columnseprule` command is not available as a key and is set to “zero” for the inner levels and the `keyans` environment. If the value of this is set inside the document, it will affect “all environments” that use the `columns` key.

1.3.4 Support for minipage

The package provides direct support for `minipage` environment, this allows you to obtain an output like the one shown in figure 5.



Figure 5: Representation of the `mini-env` output for a nested level `enumext` environment.

The `minipage` environments on “left side” and “right side” is always used with “aligned on top” [t]. It can be used in all nesting levels of the environment `enumext` and the environment `keyans` and is controlled by `mini-env` and `mini-sep` keys. In order to switch from the “left” side `minipage` environment to the “right” side one must use the command `\miniright` (see §5.6).

1.3.5 The \label and \ref system

This package provides a user interface like the `enumitem`[6] package to customize the references which is activated by the `ref` key (§5.1), the standard \TeX `\label` and `\ref` commands work as usual. It also provides an “internal reference” system for the “stored content” by means of the key `save-ref` (§6.1.1) when the key `save-ans` (§6.1) is active.

1.3.6 Support for \footnote

The `enumext*` and `keyans*` environments and the `mini-env` key use the `minipage` environment in their implementation but in a transparent way for the user, i.e. it is only used for typesetting and not directly. The `enumext` package provides an *internal implementation* for the command `\footnote` compatible with the `hyperref` package to work in the same way as if it were used anywhere in the document.

Unfortunately, if *tagging* PDF is not enabled, it will not produce the expected “links” because the internal implementation uses `\footnotetext[⟨number⟩]` and `\footnotemark[⟨number⟩]{⟨text⟩}` and support for these is limited by the `hyperref` package.

The best way to solve this if *tagged* PDF is NOT active is to use Jean-François Burnol `footnotehyper`[9] package, it will support keeping the “links” if `hyperref` is loaded with the `hyperfootnotes=true` option (default). Load it is as follows:

```
\IfDocumentMetadataTF{ }
{
  \usepackage{footnotehyper}
  \makesavenoteenv{enumext}
  \makesavenoteenv{enumext*}
}
```

At the moment the `footnotehyper` package is not compatible with *tagged* PDF.

2 The environments provided

The package `enumext` provides two main list environments, the *vertical* environment `enumext` and the *horizontal* environment `enumext*`.

| | | |
|----------|--|--|
| enumext | <code>\begin{enumext}[⟨keyval list⟩]</code> | <code>\begin{enumext*}[⟨keyval list⟩]</code> |
| enumext* | <code>\item ⟨item content⟩</code> | <code>\item ⟨item content⟩</code> |
| | <code>\item [⟨custom⟩] ⟨item content⟩</code> | <code>\item [⟨custom⟩] ⟨item content⟩</code> |
| | <code>\item* [⟨symbol⟩] [⟨offset⟩] ⟨item content⟩</code> | <code>\item* [⟨symbol⟩] [⟨offset⟩] ⟨item content⟩</code> |
| | <code>\end{enumext}</code> | <code>\end{enumext*}</code> |

2.1 The environment enumext

The `enumext` is an environment that works in the same way as the standard `enumerate` environment provided by \LaTeX , `\item` and `\item[⟨custom⟩]` commands work in the usual way. The environment can be nested with at most “four levels” and the options can be configured globally using `\setenumext` command and locally using `[⟨key = val⟩]` in the environment.

Example with `columns=2`

1. This text is in the first level.
- A. This text is in the fourth level.
- (a) This text is in the second level.
- X This text is in the first level.
- i. This text is in the third level.
- * 2. This text is in the first level.

2.2 The environment enumext*

The `enumext*` is a *horizontal list environment* similar to the `shortenurimate` or `tasks` environments provided by the `shortlst`[15] and `tasks`[16] packages, `\item` and `\item[⟨custom⟩]` work as usual. The options can be configured globally using `\setenumext` command and locally using `[⟨key = val⟩]` in the environment.

Some considerations to take into account for this environment:

- The environment cannot be nested within itself or in the environment `keyans*`, but it can be nested within `enumext` and vice versa.
- Each “item content” in the environment is placed within a `minipage` environment whose *width* is stored in the dimension `\itemwidth` that NOT includes `labelwidth`, `labelsep`, only the *width of the content*.
- You cannot have floating environments like `figure` or `table` but `\footnote` with `hyperref` support is supported if the `footnotehyper` package is loaded (see §1.3.6 for full support).
- You cannot have any standard list environments like `itemize`, `enumerate`, `description`, `quote`, `quotation`, `verse`, `center`, `flushleft`, `flushright`, `verbatim`, `tabbing`, `trivlist`, `list` and all environments created with `\newtheorem`.

Example with `columns=2`

1. This text is in the first level.
2. This text is in the first level.
- X This text is in the first level.
- * 4. This text is in the first level.

2.3 The command \item*

```
\item* \item* [⟨symbol⟩] [⟨offset⟩]
```

The `\item*`, `\item* [⟨symbol⟩]` and `\item* [⟨symbol⟩] [⟨offset⟩]` works like the numbered `\item`, but placing a `⟨symbol⟩` to the “left” of the `⟨label⟩` separated from it by the `⟨offset⟩` set by the the *second optional argument*. The *starred argument* “*” cannot be separated by spaces ‘`␣`’ from the command, i.e. `\item*` and the *first optional argument* does “NOT” support *verbatim content*. Can be configure with the keys `item-sym*` and `item-pos*` locally in the environment or globally using `\setenumext` command (§3).

The behavior of `\item*` in the `enumext` and `enumext*` environments is NOT the same as in the `keyans` and `keyans*` environments.

2.3.1 Keys for \item*

`item-sym*` = {<symbol>} default: \textasteriskcentered
Sets the *symbol* to be displayed in the “left” of the box containing the current <label> set by `labelwidth` key for `\item*` in `enumext` and `enumext*`. The *symbol* can be in *text* or *math* mode, for example `item-sym*={\star}`.
`item-pos*` = {<rigid length>} default: by levels
Sets the *offset* between the box containing the current <label> defined by `labelwidth` key and the <symbol> set by `item-sym*` key. The default values are set by `labelsep` key at each level. If positive values are passed it will *offset to the left* and if negative values are passed it will *offset to the right*.

2.4 The command \item in enumext*

The `\item` command for the `enumext*` environment provides an “first optional argument” `\item(<columns>)` which “joins items” between columns. Let’s consider the following examples adapted directly from the `task` package:

```
\begin{enumext*}[widest=10,columns=4]
  \item The first
  \item* The second
  \item The third
  \item The fourth
  \item(3)* The fifth item is way too long for this and needs three columns
  \item The sixth
  \item The seventh
  \item(2)[X] The eighth item is way too long for this and needs two columns
    (\the\itemwidth)
  \item The ninth
  \item[Z] The tenth (\the\itemwidth)
\end{enumext*}
```

1. The first
- * 2. The second
3. The third
4. The fourth
- * 5. The fifth item is way too long for this and needs three columns
6. The sixth
7. The seventh
- X 8. The eighth item is way too long for this and needs two columns (196.17749pt)
9. The ninth
- Z 10. The tenth (89.28171pt)

3 The command \setenumext

| | | |
|--------------------------|---|---|
| <code>\setenumext</code> | <code>\setenumext{<key = val>}</code> | <code>\setenumext[<keyans*>]{<key = val>}</code> |
| | <code>\setenumext[<enumext, level>]{<key = val>}</code> | <code>\setenumext[<print, level>]{<key = val>}</code> |
| | <code>\setenumext[<enumext*>]{<key = val>}</code> | <code>\setenumext[<print, *>]{<key = val>}</code> |
| | <code>\setenumext[<keyans>]{<key = val>}</code> | <code>\setenumext[<print*>]{<key = val>}</code> |

The command `\setenumext` sets the <keys> on a global basis for environments `enumext`, `enumext*`, `keyans`, `keyans*` and the `\printkeyans` command. It can be used both in the preamble and in the body of the document as many times as desired.

The <keys> set in the *optional argument* of environments and commands have the *highest precedence*, overriding both options passed by `\setenumext`. If the *optional argument* is not passed, the first level of the environment `enumext` will be taken by default.

- The key `save-ans` that activate the “storage system” must NOT be passed through this command and must be passed directly in the *optional argument* of the “first level” of the environment in which they are executed.

4 The command \setenumextmeta

| | |
|------------------------------|--|
| <code>\setenumextmeta</code> | <code>\setenumextmeta {<key name>}{<key-one = val, key-two = val, ...>}</code> |
| | <code>\setenumextmeta*{<key name>}{<key-one = val, key-two = val, ...>}</code> |
| | <code>\setenumextmeta [<enumext*>]{<key name>}{<key-one = val, key-two = val, ...>}</code> |
| | <code>\setenumextmeta [<enumext, level>]{<key name>}{<key-one = val, key-two = val, ...>}</code> |

The command `\setenumextmeta` adds a new “meta-key” for the environments `enumext` and `enumext*`, the {<key name>} must be different from those defined by the package. If the *optional argument* is not passed, the new “meta-key” will be created for the “first level” of the environment `enumext`.

The *starred argument* ‘*’ will create the new “meta-key” for the environment `enumext*` and for all levels of the environment `enumext`. For example: `\setenumextmeta*{midsep}{topsep=3pt, partopsep=0pt}` will create a new key `midsep` available for all levels of the `enumext` environment and the `enumext*` environment and we can use it like any other key so `\begin{enumext}[midsep]` and `\begin{enumext*}[midsep]` will be valid.

5 The keyval system

The $\langle key = val \rangle$ system used by the `enumext` package is implemented using `l3keys` so it must be taken into consideration that those keys marked as “*value forbidden*”, that is $\langle key \rangle$ is different from $\langle key = \rangle$.

All $\langle keys \rangle$ described in this section are available for the `enumext`, `enumext*`, `keyans` and `keyans*` environments with the exception of the keys `series`, `resume`, `resume*` which are only available for the “*first level*” of the environments `enumext` and `enumext*`; and the keys `mini-right`, `mini-right*` which are only available for the `enumext*` and `keyans*` environments.

All $\langle keys \rangle$ related to vertical or horizontal spacing accept a “*skip*” or “*dim*” expression if passed between braces, i.e. you do not need to use `\dimeval` or `\dimexpr` to perform calculations.

- It should be kept in mind that using any $\langle key \rangle$ that sets a *rubber lengths* or *rigid lengths* for vertical or horizontal space on a level will influence the vertical and horizontal space for *inners levels* and `keyans`, `keyans*` and `keyanspic` environments.

5.1 Keys for label and ref

`mode-box` $\langle value forbidden \rangle$ default: *not used*

This is a “*switch-key*” that does not receive an argument and is “*only*” available for the “*first level*” of the `enumext` environment and the `enumext*` environment. When this is set the `label`, `font`, `wrap-label` and `wrap-label*` keys are executed within `\makebox` for the `enumext` and `keyans` environments.

- This key is intended for compatibility with *tagged* PDF and is forcibly “*enabled*” when `\DocumentMetadata` is present. If you want to get the same document output whether `\DocumentMetadata` is active or not, you must enable this key.
- In the `enumext*` and `keyans*` environments `\makeLabel` are redefined using `\makebox` by default. If `enumext` or `keyans` is used in the `enumext*` environment the key must be activated manually.

`label` = { $\langle \backslash alph* | \backslash Alph* | \backslash arabic* | \backslash roman* | \backslash Roman* \rangle$ } default: *by levels*

Sets the $\langle label \rangle$ that will be printed at the *current level*. The default value for the first level of the environments `enumext` and `enumext*` are `\arabic*`, for second level are $\langle \backslash alph* \rangle$, for third level are `\roman*`, and for fourth level are `\Alph*`. For `keyans` and `keyans*` environments the default value is `\Alph*`.

- This key is intended to give the basic structure with which the $\langle label \rangle$ will be displayed, and the form in which it is used by standard “*label and ref*” and the “*internal label and ref*” system with the `save-ref` key. You cannot use commands with $\langle label \rangle$ as an argument, for example `\emph{\backslash alph*}` will return an error. For full customization of how $\langle label \rangle$ is displayed use the `font`, `wrap-label` and/or `wrap-label*` keys.

`labelsep` = { $\langle rigid length \rangle$ } default: `0.3333em`

Sets the *horizontal space* between the box containing the current $\langle label \rangle$ defined by `label` key and the text of an item on the first line. Internally sets the value of `\labelsep` for the current level.

`labelwidth` = { $\langle rigid length \rangle$ } default: *by label*

Sets the *width* of the box containing the current $\langle label \rangle$ set by `label` key. Internally sets the value of `\labelwidth` for the current level. The default values are calculated by means of the *width* of a box by setting a *value* to the current counter using ‘0’ for `\arabic*`, ‘M’ for `\Alph*`, ‘m’ for `\alph*`, ‘VIII’ for `\Roman*` and ‘viii’ for `\roman*`.

`widest` = { $\langle integer | string \rangle$ } default: *empty*

Sets the `labelwidth` key pass the $\langle integer \rangle$ or converting the $\langle string \rangle$ of the form `\Alph`, `\alph`, `\Roman` or `\roman` to a *value* for the current counter defined by `label` key, then calculating the *width* by means of a box. For example `widest={XXIII}` or `widest={23}` are equivalent. This key is useful when the default values of the `labelwidth` key are smaller than those actually used.

`font` = { $\langle font commands \rangle$ } default: *empty*

Sets the *font style* for the current $\langle label \rangle$ defined by `label` key. For example `font={\bfseries\small}`.

`align` = { $\langle left | right | center \rangle$ } default: *left*

Sets the *aligned* of $\langle label \rangle$ defined by `label` key on the current level in the label box.

`wrap-label` = { $\langle code \{ \#1 \} \text{ more code} \rangle$ } default: *empty*

Wraps the *current* $\langle label \rangle$ defined by `label` key referenced by `\{ \#1 \}`. The $\langle code \rangle$ must be passed between braces. This key does not modify the value set by the `labelwidth` key and is applied only on `\item` and `\item*`. When using it in the `\setenumext` command it is necessary to use the *double hash* ‘`\{ \# \#1 \}`’. For example `wrap-label={\fbox{\#1}}` or you can create a command:

```
\NewDocumentCommand \mywrap { s m }
{
  \IfBooleanTF{\#1}
  {
    {\textcolor{red}{\textbf{Q}}\textcolor{blue}{\textbf{.}}\textcolor{gray}{\#2}}
    {\textcolor{blue}{\textbf{Q}}\textcolor{red}{\textbf{.}}\textcolor{gray}{\#2}}
  }
}
```

and then pass it through the key `wrap-label={\mywrap{\#1}}` or `wrap-label={\mywrap*{\#1}}`.

`wrap-label*` = { $\langle code \{ \#1 \} \text{ more code} \rangle$ } default: *empty*

The same as the `wrap-label` key but also applies on `\item[custom]`.

`ref = {⟨code⟩ {⟨\alph*⟩⟨\Alph*⟩⟨\arabic*⟩⟨\roman*⟩⟨\Roman*⟩ more code⟩}` default: *empty*

Modifies the way *cross references* are displayed. The `label` key sets the default form of the *cross references*, by using this key you can define a different format, for example: `ref=\emph{⟨\alph*⟩}` is valid.

Internally it renews the command associated with each counter when it is executed, i.e., in the environment `enumext` the command `\theenumxi` is modified when the key is executed at the first level, `\theenumxii` when it is executed at the second level and `\theenumxiii` together with `\theenumxiv` when it is executed at the third and fourth levels.

- This must be kept in mind, since the values set by the `label` and `ref` keys are not cumulative by levels, so if you have used the `ref` key in the first level and then want to associate the counter with `label` or `ref` in the second level you must use the direct commands, i.e. `\arabic{enumxi}` to indicate the count of the first level instead of using `\theenumxi`.

5.2 Keys for spaces

`show-length = {⟨true⟩|⟨false⟩}` default: *false*

Displays on the terminal the values for *all list parameters* at the current level. For *vertical spaces* show the values of `\topsep`, `\itemsep`, `\parsep` and `\partopsep`. For *horizontal spaces* show the values of `\labelwidth`, `\labelsep`, `\itemindent`, `\listparindent` and `\leftmargin`.

5.2.1 Vertical spaces

`topsep = {⟨rubber length⟩|⟨rigid length⟩}` default: *by levels*

Set the *vertical space* added to both the top and bottom of the list. Internally sets the value of `\topsep` for the current level. The default value for the first level of the environments `enumext` and `enumext*` are `8.0pt` plus `2.0pt` minus `4.0pt`, for second level are `4.0pt` plus `2.0pt` minus `1.0pt`, for third and fourth level are `2.0pt` plus `1.0pt` minus `1.0pt`. For `keyans` and `keyans*` environments the default value is `4.0pt` plus `2.0pt` minus `1.0pt`.

`parsep = {⟨rubber length⟩|⟨rigid length⟩}` default: *by levels*

Set the *vertical space* between paragraphs within an item. Internally sets the value of `\parsep` for the current level. The default value for the first level of the environments `enumext` and `enumext*` are `4.0pt` plus `2.0pt` minus `1.0pt`, for second level are `2.0pt` plus `1.0pt` minus `1.0pt`, for third and fourth level are `0pt`. For `keyans` and `keyans*` environments the default value is `2.0pt` plus `1.0pt` minus `1.0pt`.

- In the `enumext*` and `keyans*` environments this value is passed to `\parskip` within the `minipage` environment where “item content” is placed.

`partopsep = {⟨rubber length⟩|⟨rigid length⟩}` default: *by levels*

Set the *vertical space* added, beyond `topsep`, to the “top” and “bottom” of the entire environment if the environment instance is preceded by a “blank line” or `\par` command. Internally sets the value of `\partopsep` for the current level. The default values for first and second level in environment `enumext` are `2.0pt` plus `1.0pt` minus `1.0pt`, for third and fourth level are `1.0pt` minus `1.0pt`. For the `keyans` environment the default value is `2.0pt` plus `1.0pt` minus `1.0pt`, and for the `keyans*` and `enumext*` environments it is available but *without* effect.

- The value of this parameter also affects the *inner levels* and the environments `keyans`, `keyanspic` and `keyans*`. Caution should be taken with “blank lines” or `\par` command “before” each environment or nested level when formatting the source code of document. T_EX will enter *⟨vertical mode⟩* and apply this value to the “top” and “bottom” the environment or nested level.

`itemsep = {⟨rubber length⟩|⟨rigid length⟩}` default: *by levels*

Set the *vertical space* between items, beyond the `parsep`. Internally sets the value of `\itemsep` for the current level. The default value for the first level of the environments `enumext` and `enumext*` are `4.0pt` plus `2.0pt` minus `1.0pt`, for the rest of the levels are `2.0pt` plus `1.0pt` minus `1.0pt`. For `keyans` and `keyans*` environments the default value is `4.0pt` plus `2.0pt` minus `1.0pt`.

- In the `enumext*` and `keyans*` environments this value corresponds to the separation between rows.

`noitemsep` *⟨value forbidden⟩* default: *not used*

This is a “meta-key” that does not receive an argument. Set `itemsep` and `parsep` equal to `0pt` the entire level of environment.

`nosep` *⟨value forbidden⟩* default: *not used*

This is a “meta-key” that does not receive an argument. Sets all keys for vertical spacing equal to `0pt` the entire level of environment.

`base-fix` *⟨value forbidden⟩* default: *not used*

This is a “switch-key” that does not receive an argument available *only* for the “first level” of environment `enumext`. Fix the *baseline* when an environment `enumext` is nested in `enumext*` and there is no material between the `\item` and the start of the environment for example `\item \begin{enumext}` within the environment `enumext*`. Internally sets the keys `topsep`, `above` and `above*` at `0pt`.

- This key is provided as a way to work around this minor issue, but you should be aware that if for some reason you have the `itemindent` key set in the `enumext*` environment it will be lost and you will need to adjust it using the `list-offset` key in the `enumext` environment.

- The following $\langle keys \rangle$ should be used with “caution”, they are intended to be used at the “top” and “bottom” of the environment when the `columns` or `mini-env` keys do not provide adequate *vertical spaces*. The values passed can be *rubber* or *rigid* lengths, the way they are applied is the way you differ, using the star ‘*’ $\langle keys \rangle$ applies `\vspace*` so that \LaTeX does *not discard* this space at page break.

`above = { $\langle rubber length \mid rigid length \rangle$ }` default: *not used*

Set the *extra vertical space* added, beyond `topsep`, to the top of the entire level of environment. This key is intended to give a “fine adjustment” of the vertical space “above” the environment without hindering the value of the `topsep` key. The space is added with `\vspace` so is “discordable”.

`above* = { $\langle rubber length \mid rigid length \rangle$ }` default: *not used*

Set the *extra vertical space* added, beyond `topsep`, to the top of the entire level of environment. This key is intended to give a “fine adjustment” of the vertical space “above” the environment without hindering the value of the `topsep` key. The space is added with `\vspace*` so is “not discordable”.

`below = { $\langle rubber length \mid rigid length \rangle$ }` default: *not used*

Set the *extra vertical space* space added, beyond `topsep`, to the bottom of the entire level of environment. This key is intended to give a “fine adjustment” of the vertical space on the “below” the environment without hindering the value of the `topsep` key. The space is added with `\vspace` so is “discordable”.

`below* = { $\langle rubber length \mid rigid length \rangle$ }` default: *not used*

Set the *extra vertical space* space added, beyond `topsep`, to the bottom of the entire level of environment. This key is intended to give a “fine adjustment” of the vertical space on the “below” the environment without hindering the value of the `topsep` key. The space is added with `\vspace*` so is “not discordable”.

5.2.2 Horizontal spaces

`list-offset = { $\langle rigid length \rangle$ }` default: `0pt`

Sets the *horizontal translation* of the entire environment level from the left edge of the box defined by the `labelwidth` key. Internally sets the values of `\leftmargin` and `\itemindent` for the current level.

`list-indent = { $\langle rigid length \rangle$ }` default: `labelwidth + labelsep`

Sets the *indentation* of the whole environment under the box defined by `labelwidth` and `labelsep` keys. Internally sets the value of `\leftmargin` and `\itemindent` for the current level. If `list-indent=0pt` is set in the environments `enumext` and `keyans` the $\langle label \rangle$ will be part of the text, separated by the value of the `labelsep` key and the *first word*, in simple terms it will look like a “common paragraph”.

- The `enumext*` and `keyans*` environments are implemented using `\makebox` and `minipage` which causes “list indent” to always be equal to the value passed to `labelwidth` plus `labelsep`. Passing a value to this key is equivalent to setting the value for the `list-offset` key.

`itemindent = { $\langle rigid length \rangle$ }` default: `0pt`

Sets the extra *horizontal indentation*, beyond `labelsep`, of the “first line” off each `\item` that is not followed by a “blank line” or the `\par` command. This value must be greater than or equal to `0pt` and is applied internally using `\hspace` without modifying the value of `\itemindent`.

- This key is intended for the `enumext*` and `keyans*` environments where, by their implementation, it is not possible to adjust `labelwidth` and `list-indent` without modifying the output. If you use `enumext` or `keyans` and want to get around the *blank line* limitation or the `\par` command followed by `\item` you can modify `labelwidth` and `list-indent` and get the same effect.

`rightmargin = { $\langle rigid length \rangle$ }` default: `0pt`

Set the *horizontal space* between the right margin of the environment and the right margin of the enclosing environment, the value it takes must be greater than or equal to `0pt`. Internally sets the value of `\rightmargin` for the current level.

`listparindent = { $\langle rigid length \rangle$ }` default: `0pt`

Sets the *horizontal space* indentation, beyond `list-indent`, for second and subsequent paragraphs within a list item. Internally sets the value of `\listparindent` for the current level.

- In the `enumext*` and `keyans*` environments this value is passed to `\parindent` within the `minipage` environment where “item content” is placed.

5.3 Keys for add code

The following $\langle keys \rangle$ should be used with “caution”, they are intended to inject $\{ \langle code \rangle \}$ into different parts of the defined environments. We must keep in mind that the defined environments are based on the `list` base environment provided by \LaTeX which is defined (simplified) as plain form `\list{ $\langle arg one \rangle$ }{ $\langle arg two \rangle$ }`. Using the `before*` key does not allow access to the `list` parameters defined by $[\langle key = val \rangle]$.

`before = { $\langle code \rangle$ }` default: *not used*

Execute $\{ \langle code \rangle \}$ “before” the environment starts. The $\{ \langle code \rangle \}$ must be passed between braces, is executed “after” performing all calculations related to the *list parameters* in the environment and the parameters sets by $[\langle key = val \rangle]$ that is, in the second argument of the list after setting all the parameters `\begin{list}{ $\langle arg one \rangle$ }{ $\langle arg two \rangle$ }{ $\langle code \rangle$ }`.

`before*` = {`<code>`} default: *not used*
 Execute {`<code>`} “before” the environment starts. The {`<code>`} must be passed between braces, is executed “before” performing all calculations related to the `list parameters` and [`<key = val>`] sets in the environment that is, before the arguments defining the environment are executed: {`<code>`}\begin{list}{`<arg one>`}{`<arg two>`}.

`first` = {`<code>`} default: *not used*
 Executes {`<code>`} when “starting” the environment. The {`<code>`} must be passed between braces, is executed right “after” all `list parameters` are done, after the second argument of list, just before the first occurrence of \item: \begin{list}{`<arg one>`}{`<arg two>`}{`<code>`}\item.

- 🔴 Keep in mind that the code set in this key will affect the entire “body” of the environment and therefore the inner levels of the list and the `keyans` environment. It is recommended to set this key per level.
- 🔴 In the `enumext*` and `keyans*` environments this key is executed after the `listparindent`, `parsep` and `itemindent` keys within the `minipage` environment in which the “item content” is placed.

`after` = {`<code>`} default: *not used*
 Execute {`<code>`} “after” finishing the environment. The {`<code>`} must be passed between braces.

5.4 Keys for start, series and resume

`start` = {`<integer | integer expression>`} default: `1`
 Sets the *start value* of the numbering on the current level. The {`<integer expression>`} must be passed between braces, internally is evaluated and pass to the counter defined by `label` key on the current level, i.e. it is equivalent to enter `start={\dimeval{100*\value{chapter}}` or `start={100*\value{chapter}}`.

`start*` = {`<integer | string>`} default: *not used*
 Sets the *start value* of the numbering on the current level. Internally `<string>` is converted and passed as value to the counter defined by `label` key on the current level, i.e. it is equivalent to enter `start=5`, `start=E` or `start=v`.

The following `<keys>` are “only” available for the `enumext*` environment and the “first level” of the `enumext` environment and are ignored if set when nested within each other.

`series` = {`<series name>`} default: *not used*
 Stores the *keys* of the *optional argument* of the “first level” of the environment in which it is executed in {`<series name>`} which is used as an argument in the key `resume`. The `<keys>` stored in {`<series name>`} are not cumulative and are overwritten if the same {`<series name>`} is used again.

`resume` = {`<series name>`} default: *not used*
 Sets the *start value* and *options* for the “first level” continuing the numbering of the environment in which the `series={<series name>}` key was executed. If passed *without value* this will only set *start value* continue the numbering from the last environment in which `series={<series name>}` or `resume={<series name>}` is not present and if the `save-ans` key is active it will continue the numbering from the last environment in which it was executed. The *start value* can be overwritten using `start` or `start*` keys.

`resume*` `<value forbidden>` default: *not used*
 Sets the *start value* and *options* for the “first level” continuing the numbering of the environment in which the `series={<series name>}` or `resume={<series name>}` keys are NOT present, if the `save-ans` key is active it will continue the numbering from the last environment in which it was executed. The *start value* can be overwritten using `start` or `start*` keys.

- 🔴 For security reasons the `series` key will never save in {`<series name>`} the keys `series`, `resume`, `resume*`, `save-ans`, `save-key`, `start*` and `start`. When using the key `resume={<series name>}` it will have hierarchy in the `<keys>` that are saved in {`<series name>`}, in order to establish the value of a `<key>` already saved in {`<series name>`} it must be placed to the “right” of `resume={<series name>}`, the same thing happens with the `resume*` key, the exception is the `save-ans` key that must be placed on the “left” if you want to start the numbering with its value. The `resume` key passed “without value” must be exactly “without value”, i.e. `resume=` cannot be used and if executed before `resume*` it will affect the *start value*.

5.5 Keys for multicols

`columns` = {`<integer>`} default: `1`
 Set the *number of columns* to be used by the `multicols` environment within the environment. The value must be a positive integer less than or equal to `10`.

`columns-sep` = {`<rigid length>`} default: *by level*
 Set the *space between columns* used by the `multicols` environment within the environment. Internally sets the value of `\columnsep`, by default its value is equal to the sum of the values set in the keys `labelwidth` and `labelsep` of the current level.

5.6 Keys for minipage

`mini-env = {⟨rigid length⟩}`

default: *not used*

Sets the *width* of the `minipage` environment on the “right side”. This value added to the value set by the `mini-sep` key to determines the *width* of the `minipage` environment on the “left side”, taking `\linewidth` as the maximum reference value.

`mini-sep = {⟨rigid length⟩}`

default: `0.3333em`

Sets the *space between* the `minipage` environment on the “left side” and the `minipage` environment on the “right side”. This separation is applied together with `\hfill`.

5.6.1 The command `\miniright`

```
\miniright \begin{enumext}[mini-env={⟨rigid length⟩}] ⟨item's before⟩ \item \miniright ⟨content⟩ \end{enumext}
\begin{enumext}[mini-env={⟨rigid length⟩}] ⟨item's before⟩ \item \miniright*⟨content⟩ \end{enumext}
```

The `\miniright` command close the `minipage` environment on the “left side” and opens the `minipage` environment on the “right side” by starting it with the `\centering` command. It must be placed “after” the last `\item` of the current environment and “before” starting the material to be placed on the “right side”.

The *starred argument* “*” inhibits the use of `\centering` command i.e. the usual L^AT_EX justification is maintained in the `minipage` on the “right side”.

5.6.2 The key `mini-right`

In the *horizontal list environments* `enumext*` and `keyans*` it is not possible to use the `\miniright` command and the `mini-right` key must be used instead.

`mini-right = {⟨content⟩}`

default: *not used*

Set the *content* for the drawing or tabular to be placed in the `minipage` environment on the “right side” by starting it with `\centering`. The `{⟨content⟩}` must be passed between braces.

`mini-right* = {⟨content⟩}`

default: *not used*

Same as above, but *without* starting with `\centering`.

6 The storage system

The entire mechanism for “*storing content*” it is activated according to `save-ans` key on the “*first level*” of `enumext` or `enumext*` environments and it is ignored if they are established when they are nested inside each other. Only when this `⟨key⟩` is “*active*” the `\anskey` command and the environments `anskey*`, `keyans`, `keyans*` and `keyanspic` are available.

| | |
|---|---|
| <pre>\begin{enumext}[save-ans={⟨store name⟩}] \item Text \anskey{answer} \item Text \begin{keyans} ... \end{keyans} \end{enumext}</pre> | <pre>\begin{enumext}[save-ans={⟨store name⟩}] \item Text \anskey{answer} \item Text \begin{keyanspic} ... \end{keyanspic} \end{enumext}</pre> |
|---|---|

By executing the key `save-ans={⟨store name⟩}` the entire “*structure*” of the environment (excluding the *first level*) including the *optional argument* passed to the inner levels or the environment nested in it, along with the `⟨content⟩` passed to `\anskey` or `anskey*`, the current `⟨labels⟩` for `\item*` and `\anspic*` in the environments `keyans`, `keyans*` and `keyanspic` will be “*stored*” in a *sequence* `{⟨store name⟩}` and at the same time will be “*stored*” (without the “*structure*” or *optional argument*) in a *prop list* `{⟨store name⟩}`.

For security reasons the *optional argument* of the inner levels or the nested environment are *filtered* by excluding all `⟨keys⟩` related to the “*storage system*” (§6.1) along with the keys `mini-env`, `mini-sep`, `mini-right`, `mini-right*`, `series`, `resume` and `resume*` when storing in *sequence* `{⟨store name⟩}` set by `save-ans` key.

6.1 Keys for storage system

The only `⟨keys⟩` available for all levels of the `enumext` environment and the `enumext*` environment are `no-store` and `save-key`, the rest of the `⟨keys⟩` described in this section must be passed directly in the *optional argument* of the “*first level*” of the environment in which the key `save-ans` is executed. The key `save-ans` should NOT be passed with the command `\setenumext`.

`save-ans = {⟨store name⟩}`

default: *not set*

Sets the *name* of the *sequence* and *prop list* in which the `{⟨contents⟩}` will be “*stored*” by `\anskey` and `anskey*` in `enumext` and `enumext*` environments and the current `⟨labels⟩` for `\item*` and `\anspic*` in the environments `keyans`, `keyans*` and `keyanspic`. If the *sequence* or *prop list* `{⟨store name⟩}` does not exist, it will be created globally and will not be *overwritten* if the key is used again.

`save-key = {⟨key list⟩}`

default: *not set*

This key *overrides* the default “*stored keys*” of the *optional argument* of the inner levels or nested environment that will be passed to the *sequence*. The `⟨key list⟩` passed to this key ignores any `⟨keys⟩` in the “*stored structure*” and must be passed between braces. For example, if we execute at a second level:

```

\begin{enumext}[save-ans={\<store name>}]
  \item Text \anskey{answer}
  \item Text
    \begin{enumext}[nosep, columns=2, save-key={columns=3}]
      ...
    \end{enumext}
\end{enumext}

```

The “*stored keys*” by default in the *sequence* $\{\langle store name \rangle\}$ would be `nosep`, `columns=2`, but using the key `save-key={columns=3}` will overwrite and the “*stored key*” in the *sequence* $\{\langle store name \rangle\}$ are only `columns=3` ignoring all the others.

`save-sep = {\<text symbol>}` default: {,}

Sets the *text symbol* that will separate the current $\langle label \rangle$ to the *optional argument* passed to the `\item*` and `\anspic*` in the environments `keyans`, `keyans*` and `keyanspic` and storing them in the *sequence* and *prop list* $\{\langle store name \rangle\}$ set by `save-ans` key. The $\{\langle text symbol \rangle\}$ must always be passed between braces, whitespace ‘`␣`’ is preserved within the braces and only affects the “*stored content*” and not what is displayed when using the `show-ans` or `show-pos` keys.

6.1.1 Keys for label and ref

`save-ref = {\<true | false>}` default: false

Activates the “*internal label and ref*” mechanism for referencing “*stored content*” in *prop list* $\{\langle store name \rangle\}$ set by `save-ans` key. To reference the location of the “*stored content*” within the environment you must use `\ref{\<store name : position>}`, where $\langle position \rangle$ corresponds to the position occupied by the “*stored content*” in the *prop list* $\{\langle store name \rangle\}$ returned by the `show-pos` key. For example `\ref{test:4}` will return `3`. (b) which corresponds to the location of the “*stored content*” at position `4` in *prop list* `test` within the environment in which the key `save-ans=test` was set.

`mark-ref = {\<symbol>}` default: \%

Sets the *symbol* that will be displayed by the `\printkeyans` command only if the `hyperref` package is detected and the `save-ref` key are active. This “*symbol*” is used as a “*link*” between the environment in which the `save-ans` key was used and the place where the command is executed.

6.1.2 Keys for wrap and display

`wrap-ans = {\<code {#1} more code>}` default: \fbox+\parbox{#1}

Wraps the *argument* passed to the `\anskey` and the *body* in `anskey*` environment referenced by $\{#1\}$ when using the `show-ans` or `show-pos` keys. The $\{\langle code \rangle\}$ must be passed between braces and only affects the *argument* or *body* and NOT the “*stored content*” in the *sequence* and *prop list* $\{\langle store name \rangle\}$ set by `save-ans` key. If this key is passed using `\setenumext` it is necessary to use double ‘ $\{\langle \#1 \rangle\}$ ’.

`wrap-opt = {\<code {#1} more code>}` default: [{#1}]

Wraps the *optional argument* passed to the `\item*` and `\anspic*` referenced by $\{#1\}$ in the `keyans`, `keyans*` and `keyanspic` environments when using the `show-ans` or `show-pos` keys. The $\{\langle code \rangle\}$ must be passed between braces and only affects the current *optional argument* and NOT the “*stored content*” in the *sequence* and *prop list* $\{\langle store name \rangle\}$ set by `save-ans` key. If this key is passed using `\setenumext` it is necessary to use double ‘ $\{\langle \#1 \rangle\}$ ’.

`show-ans = {\<true | false>}` default: false

Displays the *argument* passed to the `\anskey`, the *body* for `anskey*` environment, the $\langle label \rangle$ for `\item*` and `\anspic*` at the place where it is executed. If the *optional argument* is present in `\item*` or `\anspic*` it will be shown using `wrap-opt` key.

`mark-ans = {\<symbol>}` default: \textasteriskcentered

Sets the *symbol* to be displayed in the left margin for `\anskey`, `anskey*`, `\item*` and `\anspic*` in the place where they are executed when using the key `show-ans`.

`mark-pos = {\<left | right>}` default: left

Sets the *aligned* of the symbol defined by `mark-ans` key. The “*symbol*” is aligned in a box with the same dimensions of the label box defined by `labelwidth` key on the current level and separated by the value of the `labelsep` key.

6.1.3 Keys for debug and checking

`show-pos = {\<true | false>}` default: false

Displays the *position* occupied by the “*stored content*” by `\anskey`, `anskey*`, `\item*` and `\anspic*` in the *prop list* $\{\langle store name \rangle\}$ set by `save-ans` key. This position is used by the `\getkeyans` command and by the `\ref` command if the `save-ref` key is active.

`check-ans = {\<true | false>}` default: false

Enables the *checking answer* mechanism displaying an appropriate message on the terminal. This key works under the logic that each `\item` or `\item*` that does not open an inner level or nested environment contains “*only one answer*” or “*only one execution*” of the `\anskey` or `anskey*`. It is intended to be used in conjunction with the `no-store` key.

no-store *<value forbidden>* default: *not used*

This is a “switch-key” that does not receive an argument and disables the “stored structure” in the *sequence* $\{\langle store\ name\rangle\}$ set by *save-ans* key at the entire level or a nested environment in which it runs. This key is intended for use in internal levels or nested *enumext* or *enumext** environments in which you want to use *enumext* or *enumext** but “without” using the *\anskey*, “without” use *anskey**, “without” interfering with the *check-ans* key and “without” storing an unwanted structure in the *sequence* $\{\langle store\ name\rangle\}$.

6.2 The command *\anskey*

\anskey *\anskey*[*<keys>*]{*<content>*}

The command *\anskey* takes a mandatory non empty argument $\{\langle content\rangle\}$ and “stores” it in the *sequence* and *prop list* $\{\langle store\ name\rangle\}$ set by *save-ans* key. By design the command cannot be nested or passed *verbatim material* in the argument and it is assumed that each *numbered \item* or *\item** within the environment in which it is active it has a “single execution” of *\anskey* unless *\item* or *\item** open a nested level or use the *no-store* key.

If *save-ref* key are active and the *hyperref*[8] package is detected, *\hyperlink* and *\hypertarget* will be used, otherwise the usual “label and ref” system provided by *LaTeX* will be used.

The *\anskey* command is available for all levels of the *enumext* environment and the *enumext** environment, but is disabled for the *keyans*, *keyans** and *keyanspic* environments.

6.2.1 Keys for *\anskey*

By default the $\{\langle content\rangle\}$ passed to *\anskey* when “storing” in the *sequence* $\{\langle store\ name\rangle\}$ has the form *\item <content>*, the following *<keys>* allow modifying the way in which it is “stored” in the *sequence*.

break-col *<value forbidden>* default: *not used*

Stores $\{\langle content\rangle\}$ in the *sequence* $\{\langle store\ name\rangle\}$ of the form *\columnbreak \item <content>*.

item-join= $\{\langle columns\rangle\}$ default: *not set*

Set the *number of columns* to be used for *\item(<columns>)* and stores $\{\langle content\rangle\}$ in the *sequence* $\{\langle store\ name\rangle\}$ of the form *\item(<columns>) <content>*.

item-star *<value forbidden>* default: *not used*

Stores $\{\langle content\rangle\}$ in the *sequence* $\{\langle store\ name\rangle\}$ of the form *\item* <content>*.

item-sym* $\{\langle symbol\rangle\}$ default: *not set*

Sets the *symbol* for *\item** when using the key *item-star* and stores $\{\langle content\rangle\}$ in the *sequence* $\{\langle store\ name\rangle\}$ of the form *\item*[\langle symbol\rangle] <content>*. The *symbol* can be in text or math mode, for example *item-sym*={\\$ \ast \\$}* stores *\item*[\\$ \ast \\$] <content>*.

item-pos* $\{\langle rigid\ length\rangle\}$ default: *not set*

Sets the *offset* for *\item** when using the keys *item-star* and *item-sym** and stores $\{\langle content\rangle\}$ in the *sequence* $\{\langle store\ name\rangle\}$ of the form *\item*[\langle symbol\rangle][\langle offset\rangle] <content>*.

Example

```
\begin{enumext}[save-ans=test,show-ans=true]
  \item* Text containing our instructions or questions. \anskey{\first answer}
  \item Text containing our instructions or questions.
    \begin{enumext}
      \item Question.\anskey{\second answer}
    \end{enumext}
  \item Text containing our instructions or questions. \anskey{\third answer}
  \item Text containing our instructions or questions. \anskey{\fourth answer}
\end{enumext}
```

- * 1. Text containing our instructions or questions.

*

first answer

2. Text containing our instructions or questions.

(a) Question.

*

second answer

3. Text containing our instructions or questions.

*

third answer

4. Text containing our instructions or questions.

*

fourth answer
- 6.3 The environment *anskey**
- anskey** *\begin{anskey*}[\langle key = val\rangle] \langle body content\rangle \end{anskey*}*
- The environment *anskey** takes a mandatory $\{\langle body\ content\rangle\}$ and “stores it” in the *sequence* and *prop list* $\{\langle store\ name\rangle\}$ set by *save-ans* key. If *save-ref* key are active and the *hyperref*[8] package is detected *\hyperlink* and *\hypertarget* will be used, otherwise the usual “label and ref” system provided by *LaTeX* will be used. By design the environment cannot be nested but full supports “verbatim material” in the body and it is assumed that each *numbered \item* or *\item** within the environment in which it is active it has a “single execution” unless *\item* or *\item** open a nested level or use the *no-store* key.
- ©2024 by Pablo González L
- 13 / 158

The `anskey*` environment is implemented using the `scontents` package, for the correct operation `\begin{anskey*}` and `\end{anskey*}` must be in different lines, all `\keys` must be passed separated by commas and “without separation” of the start of the environment. Comments “%” or “any character” after `\begin{anskey*}` or `[\key = val]` on the same line are NOT supported, the package `scontents` will return an “error” message if this happens. In a similar way comments “%” or “any character” after `\end{anskey*}` on the same line the package `scontents` will return a “warning” message.

6.3.1 Keys for anskey*

The `anskey*` environment uses the same `\keys` as the `\anskey` command next to the keys inherited from package `scontents`. The environment is available for all levels of the `enumext` environment and the `enumext*` environment, but it is disabled for the `keyans`, `keyans*` and `keyanspic` environments.

`write-env = { \file.ext }` default: not used

Sets the name of the `\external file` in which the `\contents` of the environment will be written. The `\file.ext` will be created in the working directory, relative or absolute paths are not supported. If `\file.ext` does not exist, it will be created or overwritten if the `overwrite` key is used.

`overwrite = { \true | false }` default: false

Sets whether the `\file.ext` generated by `write-env` from the `anskey*` environment will be rewritten.

`force-eol = { \true | false }` default: false

Sets if the `end of line` for the `\stored content` is hidden or not. This key is necessary only if the last line is the closing of some environment defined by the `fancyvrb` package as `\end{Verbatim}` or another environment that does not support a comments “%” after closing `\end{Verbatim}%`.

- For security reasons the keys `store-env`, `print-env` and `write-out` they have been left disabled. It is recommended that you review the `scontents`[4] documentation to understand how the keys described here work.

Example

```
\begin{enumext}[save-ans=test,show-pos=true,start=5]
  \item* Text containing our instructions or questions.
    \begin{anskey*}[item-star]
      \first answer
    \end{anskey*}

  \item Text containing our instructions or questions.

  \begin{enumext}
    \item Question.
      \begin{anskey*}
        \second answer
      \end{anskey*}
    \end{enumext}

  \item Text containing our instructions or questions.
    \begin{anskey*}
      \third answer
    \end{anskey*}

  \item Text containing our instructions or questions.
    \begin{anskey*}
      \fourth answer
    \end{anskey*}
\end{enumext}
```

- | | |
|---|---|
| * 5. Text containing our instructions or questions. | 7. Text containing our instructions or questions. |
| [5] First answer with verbatim | [7] third answer |
| 6. Text containing our instructions or questions. | 8. Text containing our instructions or questions. |
| (a) Question. | [8] fourth answer |
| [6] second answer | |

6.4 The environments keyans and keyans*

| | |
|----------------------|---|
| <code>keyans</code> | <code>\begin{keyans}[\key = val] \item \item[\custom] \item* \item*[\content] \end{keyans}</code> |
| <code>keyans*</code> | <code>\begin{keyans*}[\key = val] \item \item[\custom] \item* \item*[\content] \end{keyans*}</code> |

The `keyans` and `keyans*` environments are “enumerated list” environments designed for “multiple choice” questions activated by the `save-ans` key. This environments can NOT be nested and must always be at the “first level” of the `enumext` environment, the commands `\item` and `\item[\custom]` work in the usual and the command `\item(\columns)` is available for the `keyans*` environment.

- The behavior of `\item*` in `keyans` and `keyans*` environments is NOT the same as in the `enumext` or `enumext*` environments.
- ©2024 by Pablo González L

```
\begin{enumext}[save-ans=test]
  \item <item content>
    \begin{keyans}[<key = val>]
      \item <item content>
      \item [<custom>] <item content>
      \item* <item content>
      \item* [<content>] <item content>
    \end{keyans}
\end{enumext}
```

```
\begin{enumext}[save-ans=test]
  \item <item content>
    \begin{keyans*}[<key = val>]
      \item <item content>
      \item [<custom>] <item content>
      \item* <item content>
      \item* [<content>] <item content>
    \end{keyans*}
\end{enumext}
```

The `<keys>` set in the *optional argument* of the environment are the same (almost) as those of the `enumext` and `enumext*` environments and have *higher precedence* than those set by `\setenumext[<keyans>]{<key = val>}` or `\setenumext[<keyans*>]{<key = val>}`. If the *optional argument* is not passed or the `<keys>` are not set by `\setenumext`, the default values will be the same as the “second level” of the `enumext` environment with the difference in the `<label>` which will be set to `label=Alph*`.

6.4.1 The `\item*` in `keyans` and `keyans*`

```
\item* \item*
\item* [<content>]
```

The `\item*` and `\item* [<content>]` command “store” the current `<label>` set by `label` key next to the *optional argument* `<content>` in *sequence and prop list* `{<store name>}` set by `save-ans` key in the “first level” of the `enumext` or `enumext*` environments.

The *starred argument* ‘`*`’ cannot be separated by spaces ‘`_`’ from the command, i.e. `\item*` and the *optional argument* does “NOT” support *verbatim content*. By design it is assumed that the `\item*` will only appear “once” within the environment.

Example

```
\begin{enumext}[save-ans=test,columns=2,show-ans=true]
  \item Text containing a question.

  \begin{keyans*}[nosep,columns=2]
    \item Choice
    \item* Correct choice
    \item Choice
    \item Choice
    \item Choice
  \end{keyans*}

  \item Text containing a question and image.

  \begin{keyans}[nosep,mini-env={0.4\linewidth}]
    \item Choice
    \item Choice
    \item Choice
    \item Choice
    \item* [<note>] Correct choice
    \miniright
    \includegraphics[scale=0.25]{example-image-a}
    Some text
  \end{keyans}
\end{enumext}
```

1. Text containing a question.

A) Choice

* B) Correct choice

C) Choice

D) Choice

E) Choice
2. Text containing a question and image.


A) Choice

B) Choice

C) Choice

D) Choice

* E) [note] Correct choice



Some text
- 6.5 The environment `keyanspic`
- ```
keyanspic \begin{keyanspic}[<key = val>] \anspic* [<content>]{<drawing or tabular>} \end{keyanspic}
```
- The `keyanspic` environment is an “enumerated list” environment activated by the `save-ans` key that has the same configuration for “spacing” and `<label>` as the `keyans` environment that uses the `\anspic` command instead of `\item`. It is intended for placing *drawings or tabular* with `<label>` centered *above* or *below* in a *single line* or *upper and lower* layout style.

When the `keyanspic` environment is used *without keys* the `<labels>` are centered *below* the *drawings or tabular* in a *single line* layout style.
- ©2024 by Pablo González L
- 15 / 158

A representation of the output can be seen in the figure 6.



Figure 6: Representation of the `keyanspic` environment with `layout-sty={⟨3, 2⟩}` in `enumext`.

This environment cannot be nested and must always be at the “first level” of the `enumext` environment, the `\item` command is disabled and keys cannot be set using `\setenumext`.

### 6.5.1 Keys for `keyanspic`

`label-pos = {⟨above | below⟩}` default: *below*

Set the *position* of `⟨label⟩` to be centered “above” or “below” *drawings or tabular* when the `\anspic` command is executed.

`label-sep = {⟨rubber length | rigid length⟩}` default: *internal adjustment*

Set the *vertical spacing* between the `⟨label⟩` centered “above” or “below” and *drawings or tabular* when running the `\anspic` command.

`layout-sty = {⟨n° upper , n° lower⟩}` default: *not set*

Set the *number of drawings or tabular* that will be distributed “upper” and “lower” within the environment when executing the `\anspic` command. The value must be passed in braces and if not set or the `⟨n° lower⟩` is omitted the *drawings or tabular* will be put on a *single line*.

`layout-sep = {⟨rubber length | rigid length⟩}` default: *adjusted parsep from keyans*

Set the *vertical separation* between the number of *drawings or tabular* placed at the “upper” and “lower” within the environment when executing the `\anspic` command. Internally adjusts the `parsep` value taken from the `keyans` environment.

`layout-top = {⟨rubber length | rigid length⟩}` default: *adjusted topsep from keyans*

Set the *vertical space* added to both the top and bottom of the environment. Internally adjust the value of `topsep` taken from `keyans` environment.

### 6.5.2 The command `\anspic`

---

`\anspic` `\anspic{⟨drawing or tabular⟩}`  
`\anspic* [⟨content⟩]{⟨drawing or tabular⟩}`

The `\anspic` command take three arguments, the *starred argument* ‘`*`’ store the current `⟨label⟩` next to the *optional argument* `⟨content⟩` in *sequence* and *prop list* `{⟨store name⟩}` set by `save-ans` key.

The *starred argument* ‘`*`’ cannot be separated by spaces ‘`␣`’ from the command, i.e. `\anspic*` and the *optional argument* does “NOT” support *verbatim content*. By design it is assumed that the *starred argument* ‘`*`’ will only appear “once” within the environment.

#### Example

```
\begin{enumext}[save-ans=test,show-ans,nosep]
 \item Question with images and labels below.

 \begin{keyanspic}[layout-sty={3,2}]
 \anspic{\includegraphics[scale=0.15]{example-image-a}}
 \anspic{\includegraphics[scale=0.15]{example-image-b}}
 \anspic{\includegraphics[scale=0.15]{example-image-a}}
 \anspic{\includegraphics[scale=0.15]{example-image-a}}
 \anspic*[note]{\includegraphics[scale=0.15]{example-image-a}}
 \end{keyanspic}


 \item Question with images and labels above.


 \begin{keyanspic}[label-pos=above, layout-sty={3,2}, layout-sep=0.25cm]
 \anspic{\includegraphics[scale=0.15]{example-image-a}}
 \anspic{\includegraphics[scale=0.15]{example-image-b}}
 \anspic{\includegraphics[scale=0.15]{example-image-a}}
 \anspic{\includegraphics[scale=0.15]{example-image-a}}
 \anspic*[note]{\includegraphics[scale=0.15]{example-image-a}}
 \end{keyanspic}
```


```
\item Question with images and labels below.


\begin{keyanspic}
\anspic{\includegraphics[scale=0.15]{example-image-a}}
\anspic{\includegraphics[scale=0.15]{example-image-b}}
\anspic{\includegraphics[scale=0.15]{example-image-a}}
\anspic{\includegraphics[scale=0.15]{example-image-a}}
\anspic*[note]{\includegraphics[scale=0.15]{example-image-a}}
\end{keyanspic}
\end{enumext}
```


1. Question with images and labels below.

  
A)


  
B)

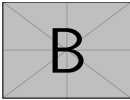
  
C)


  
D)


  
\* E) [note]


2. Question with images and labels above.

A)  



B)  


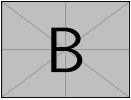
C)  



D)  



\* E) [note]  



3. Question with images and labels below on a single line.

  
A)

  
B)

  
C)

  
D)

  
\* E) [note]

Remember to pass the `alt={description}` key to the `\includegraphics` command when creating a *tagged* PDF.

6.6 Printing stored content

6.6.1 The command `\getkeyans`

```
\getkeyans \getkeyans{<store name> : <position>}
```

The command `\getkeyans` prints the “stored content” in *prop list* `{<store name>}` defined by `save-ans` key in the `<position>` returned by the `show-pos` key. The “stored content” can only be accessed *after* it is stored, if `{<store name>}` does not exist the command will return an error.

The form taken by the argument `{<store name> : <position>}` is the same as that used to generate the “internal label and ref” system when `save-ref` key are active, so to refer to a “stored content”. For example `\getkeyans{test:4}` will return the “stored content” at position `4` of the environment in which the key `save-ans=test` was set.

6.6.2 The command `\foreachkeyans`

```
\foreachkeyans \foreachkeyans[<key = val>]{<store name>}
```

The command `\foreachkeyans` goes through and executes the command `\getkeyans` on the contents in *prop list* `{<store name>}`. If you pass without options run `\getkeyans` on all contents in *prop list* `{<store name>}`.

Options for command

`sep = {<code>}` default: `{;}`

Establishes the *separation* between “each” `{<content>}` stored in *prop list* `{<store name>}`. For example, you can use `sep={\[\ 10pt]}` for vertical separation of stored contents.

`step = {<integer>}` default: `1`

Sets the *step* (increment) applied to the value set by key `start` for “each” `{<content>}` stored in *prop list* `{<store name>}`. The value must be a *positive integer*.

|                                                                                                                                                                                                                         |                |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------|
| <code>start = {⟨integer⟩}</code>                                                                                                                                                                                        | default: 1     |
| Sets the <i>position</i> of the <i>prop list</i> {⟨store name⟩} from which execution will start. The value must be a ⟨positive integer⟩.                                                                                |                |
| <code>stop = {⟨integer⟩}</code>                                                                                                                                                                                         | default: 0     |
| Sets the <i>position</i> of the <i>prop list</i> {⟨store name⟩} from which execution will finish. The value must be a ⟨positive integer⟩.                                                                               |                |
| <code>before = {⟨code⟩}</code>                                                                                                                                                                                          | default: empty |
| Sets the {⟨code⟩} that will be executed ⟨before⟩ each {⟨content⟩} stored in <i>prop list</i> {⟨store name⟩}. The {⟨code⟩} must be passed between braces.                                                                |                |
| <code>after = {⟨code⟩}</code>                                                                                                                                                                                           | default: empty |
| Sets the {⟨code⟩} that will be executed ⟨after⟩ each {⟨content⟩} stored in <i>prop list</i> {⟨store name⟩}. The {⟨code⟩} must be passed between braces.                                                                 |                |
| <code>wrapper = {⟨code {#1} more code⟩}</code>                                                                                                                                                                          | default: empty |
| Wraps the {⟨content⟩} stored in <i>prop list</i> {⟨store name⟩} referenced by {#1}. The {⟨code⟩} must be passed between braces. For example <code>\foreachkeyans[wrapper={\makebox[1em][l]{#1}}]{⟨store name⟩}</code> . |                |

### 6.6.3 The command `\printkeyans`

---

```

\printkeyans {⟨store name⟩}
\printkeyans [⟨keys⟩] {⟨store name⟩}
\printkeyans * [⟨keys⟩] {⟨store name⟩}

```

---

The command `\printkeyans` prints “all stored content” in *sequence* {⟨store name⟩} defined by `save-ans` key placing this inside the `enumext` or `enumext*` environment if the *starred argument* ‘\*’ is used.

The “stored content” can only be accessed *after* it is stored in the *sequence*, if {⟨store name⟩} does not exist the command will return an error.

The *optional argument* allows managing the ⟨keys⟩ in the “first level” of the environment in which the “stored content” of the *sequence* {⟨store name⟩} will be printed, if the *starred argument* ‘\*’ is used it will be `enumext*` otherwise `enumext`.

The default values for the “first level” are the same as the default values for the `enumext` and `enumext*` environments along with the keys `nosep`, `first=\small`, `font=\small` and `columns=2`. For the inner levels of the environment `enumext` saved in the *sequence* {⟨store name⟩} the default values are the same as those established for the second, third and fourth levels plus the keys `nosep`, `first=\small`, `font=\small`. If the environment `enumext*` is saved within the *sequence* {⟨store name⟩} it will have the same default values plus the keys `nosep`, `first=\small`, `font=\small`.

Since the command encapsulates by default the `enumext` environment or the `enumext*` environment, we must take some considerations:

- If we execute `\printkeyans*{⟨store name⟩}` and the *sequence* {⟨store name⟩} already contains any `enumext*` environment an error will be returned as we cannot nest.
- If we execute `\printkeyans*{⟨store name⟩}` and the *sequence* {⟨store name⟩} contains any `enumext` environments, they will start with the ⟨keys⟩ set for the first level unless they are set in the *optional argument* or *save-key* is used to modify it.
- If we execute `\printkeyans{⟨store name⟩}` and the *sequence* {⟨store name⟩} contains any environment `enumext*`, they will start with the ⟨keys⟩ set by default unless they are set in the *optional argument* or *save-key* is used to modify it.

The default values for the “first level” of `\printkeyans` commands and `\printkeyans*` are established using `\setenumext[⟨print, 1⟩]{⟨keys⟩}` and `\setenumext[⟨print*⟩]{⟨keys⟩}`.

If we need to set the ⟨keys⟩ for the environment `enumext` “saved” in the *sequence* {⟨store name⟩} we will use `\setenumext[⟨print, level⟩]{⟨keys⟩}` and if we need to set the ⟨keys⟩ for the environment `enumext*` “saved” in the *sequence* {⟨store name⟩} we will use `\setenumext[⟨print, *⟩]{⟨keys⟩}`.

#### Example

```

\begin{enumext}[save-ans=sample,columns=1,show-pos=true,nosep,save-ref=true]
 \item Factor $3x+3y+3z$. \anskey{$3(x+y+z)}$
 \item True False

 \begin{enumext}[nosep]
 \item \LaTeX2e\ is cool? \anskey{Very True!}
 \end{enumext}

 \item Related to Linux

 \begin{enumext}[nosep]
 \item You use linux? \anskey{Yes}
 \end{enumext}
\end{enumext}

```



```
\item Rate the following package and class
\begin{enumext}[nosep]
 \item \texttt{xsim} \anskey{very good}
 \item \texttt{exsheets} \anskey{obsolete}
\end{enumext}
\end{enumext}
\end{enumext}
```

The answer to `\ref{sample:4}` is `\getkeyans{sample:4}` and the answers to all the worksheets are as follows:

```
\printkeyans{sample}
```

1. Factor  $3x + 3y + 3z$ .

[1]

2. True False

(a)  ~~True~~ is cool?

[2]

3. Related to Linux

(a) You use linux?

[3]

(b) Rate the following package and class

i. `xsim`

[4]

ii. `exsheets`

[5]

The answer to 3.(b).i is very good and the answers to all the worksheets are as follows:

1.  $3(x + y + z)$  ✖
2. (a) Very True! ✖
3. (a) Yes ✖
- (b) i. very good ✖
- ii. obsolete ✖

7 Full examples

Here I will leave as an example some adaptations questions taken from [TeX-SX](#). The examples are attached to this documentation and can be extracted from your PDF viewer or from the command line by running:

```
$ pdftdetach -saveall enumext.pdf
```

and then you can use the excellent [arara](#)<sup>1</sup> tool to compile them.

Example 1

Adapted from the response given by Enrico Gregorio in [Squares for answer choice options and perfect alignment to mathematical answers](#) .

1. La velocità di  $1,00 \times 10^2$  m/s espressa in km/h è:

☐ A 36 km/h.

☐ B 360 km/h.

☐ C 27,8 km/h.

☐ D  $3,60 \times 10^8$  km/h.

3. La velocità di  $1,00 \times 10^2$  m/s espressa in km/h è:

☐ A 36 km/h.

☐ B 360 km/h.

☐ C 27,8 km/h.

☐ D  $3,60 \times 10^8$  km/h.

2. In fisica nucleare si usa l'angstrom (simbolo:  $1 \text{ \AA} = 1 \times 10^{-10} \text{ m}$ ) e il fermi o femtometro ( $1 \text{ fm} = 1 \times 10^{-15} \text{ m}$ ). Qual è la relazione tra queste due unità di misura?

☐ A  $1 \text{ \AA} = 1 \times 10^5 \text{ fm}$ .

☐ B  $1 \text{ \AA} = 1 \times 10^{-5} \text{ fm}$ .

☐ C  $1 \text{ \AA} = 1 \times 10^{-15} \text{ fm}$ .

☐ D  $1 \text{ \AA} = 1 \times 10^3 \text{ fm}$ .

4. In fisica nucleare si usa l'angstrom (simbolo:  $1 \text{ \AA} = 1 \times 10^{-10} \text{ m}$ ) e il fermi o femtometro ( $1 \text{ fm} = 1 \times 10^{-15} \text{ m}$ ). Qual è la relazione tra queste due unità di misura?

☐ A  $1 \text{ \AA} = 1 \times 10^5 \text{ fm}$ .

☐ B  $1 \text{ \AA} = 1 \times 10^{-5} \text{ fm}$ .

☐ C  $1 \text{ \AA} = 1 \times 10^{-15} \text{ fm}$ .

☐ D  $1 \text{ \AA} = 1 \times 10^3 \text{ fm}$ .


1. B

2. A

3. B

4. A

Example 2

Adapted from the response given by Florent Rougon in [Multiple choice questions with proposed answers in random order — addition of automatic correction \(cross mark\)](#) .

<sup>1</sup>The cool TeX automation tool: <https://www.ctan.org/pkg/arara>

©2024 by Pablo González L

19 / 158

1. La velocità di  $1,00 \times 10^2$  m/s espressa in km/h è:
- ☐ A 36 km/h.  
☒ B 360 km/h.  
☐ C 27,8 km/h.  
☐ D  $3,60 \times 10^8$  km/h.
2. In fisica nucleare si usa l'angstrom (simbolo:  $1 \text{ \AA} = 1 \times 10^{-10}$  m) e il fermi o femtometro ( $1 \text{ fm} = 1 \times 10^{-15}$  m). Qual è la relazione tra queste due unità di misura?
- ☒ A  $1 \text{ \AA} = 1 \times 10^5 \text{ fm}$ .  
☐ B  $1 \text{ \AA} = 1 \times 10^{-5} \text{ fm}$ .  
☐ C  $1 \text{ \AA} = 1 \times 10^{-15} \text{ fm}$ .  
☐ D  $1 \text{ \AA} = 1 \times 10^3 \text{ fm}$ .
3. La velocità di  $1,00 \times 10^2$  m/s espressa in km/h è:
- ☐ A 36 km/h.  
☒ B 360 km/h.  
☐ C 27,8 km/h.  
☐ D  $3,60 \times 10^8$  km/h.
4. In fisica nucleare si usa l'angstrom (simbolo:  $1 \text{ \AA} = 1 \times 10^{-10}$  m) e il fermi o femtometro ( $1 \text{ fm} = 1 \times 10^{-15}$  m). Qual è la relazione tra queste due unità di misura?
- ☒ A  $1 \text{ \AA} = 1 \times 10^5 \text{ fm}$ .  
☐ B  $1 \text{ \AA} = 1 \times 10^{-5} \text{ fm}$ .  
☐ C  $1 \text{ \AA} = 1 \times 10^{-15} \text{ fm}$ .  
☐ D  $1 \text{ \AA} = 1 \times 10^3 \text{ fm}$ .
1. B  
 3. B
- ✱ 2. A  
 ✱ 4. A

### Example 3

A “simple multiple choice” test .

1. First type of questions

- A) value  
 C) value

- B) correct  
 D) value

2. Second type of questions

- I.  $2\alpha + 2\delta = 90^\circ$   
 II.  $\alpha = \delta$   
 III.  $\angle EDF = 45^\circ$

- ☐ A I only  
☐ B II only  
☐ C I and II only

- ☐ D I and III only  
☐ E I, II, and III

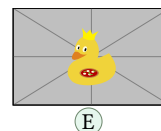
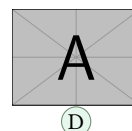
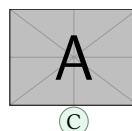
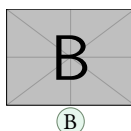
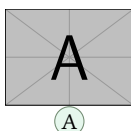
3. Third type of questions

- (1)  $2\alpha + 2\delta = 90^\circ$   
 (2)  $\angle EDF = 45^\circ$

- ☐ A value  
☐ B value  
☐ C value

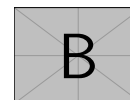
- ☐ D value  
☐ E value

4. Question with image and label below:



5. Question with image on right side:

- ☐ A value  
☐ B value  
☐ C value  
☐ D correct  
☐ E value




Test keys

1. B),  $x = 5$   
 2. D  
 3. C, some note

- ✱ 4. E, A duck  
 ✱ 5. D, other note  
 ✱

### Example 4

A “simple worksheet” using ducks :) .



Factor  $x^2 - 2x + 1$



Factor  $3x + 3y + 3z$

The following questions need to be cuaqtified :)



True False

- (a)  $\alpha > \delta$   
 (b)  $\text{\LaTeX}$  is cool?



Related to Linux

- (a) You use linux?  
 (b) Usually uses the package manager?

- (c) Rate the following package and class
- i. xsim-exam
  - ii. xsim
  - iii. exsheets

The answer to 1 is  $(x - 1)^2$  and the answer to 3.(a) is False.

- |                   |   |                                 |   |
|-------------------|---|---------------------------------|---|
| 1. $(x - 1)^2$    | ⌘ | (b) Yes, dnf                    | ⌘ |
| 2. $3(x + y + z)$ | ⌘ | (c) i. doesn't exist for now :( | ⌘ |
| 3. (a) False      | ⌘ | ii. very good                   | ⌘ |
| (b) Very True!    | ⌘ | iii. obsolete                   | ⌘ |
| 4. (a) Yes        | ⌘ |                                 |   |

Example 5

Adapted from the response given by Stephen in SAT like question format .

|                                                                                                                                                                                                                                                                                                                                                                                                                            |                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <div>1</div> <p>Which choice best describes what happens in the passage?</p> <p>A) One character argues with another character who intrudes on her home.</p> <p>B) One character receives a surprising request from another character.</p> <p>C) One character reminisces about choices she has made over the years.</p> <p>D) One character criticizes another character for pursuing an unexpected course of action.</p> | <div>3</div> <p>Which choice best describes what happens in the passage?</p> <p>A) One character argues with another character who intrudes on her home.</p> <p>B) One character receives a surprising request from another character.</p> <p>C) One character reminisces about choices she has made over the years.</p> <p>D) One character criticizes another character for pursuing an unexpected course of action.</p> |
| <div>2</div> <p>Which choice best describes what happens in the passage?</p> <p>A) One character argues with another character who intrudes on her home.</p> <p>B) One character receives a surprising request from another character.</p> <p>C) One character reminisces about choices she has made over the years.</p> <p>D) One character criticizes another character for pursuing an unexpected course of action.</p> | <div>4</div> <p>Which choice best describes what happens in the passage?</p> <p>A) One character argues with another character who intrudes on her home.</p> <p>B) One character receives a surprising request from another character.</p> <p>C) One character reminisces about choices she has made over the years.</p> <p>D) One character criticizes another character for pursuing an unexpected course of action.</p> |









1. A)                      2. C)                      3. B)                      4. D)





8 Tagged PDF examples

This section is just to show the compatibility of enumext with tagged PDF using lualatex. The attached files here are just for testing and are intended as examples and, in a way, to simplify the time of Matthew Bertucci (@mbertucci) when he sees this excellent package and adds it to The LaTeX Tagged PDF repository.

To compile the tests with lualatex-dev the packages multicol, scontents, unicode-math, geometry, graphicx, luamml and hyperref are required along with the line:

```
\DocumentMetadata
{
 lang = en-US, pdfversion = 2.0, pdfstandard = ua-2,
 testphase = {phase-III, math, title, table, firstaid},
}
```

- ◆ All examples have been checked using veraPDF together with ngpdf.
- The file enumext-01.tex contains the basic tests for the enumext and enumext\* environments and the nesting between them plus the use of the label, labelwidth, labelsep, ref, align and wrap-label keys. Source file  and tagged PDF .
  - The file enumext-02.tex contains the tests for the enumext and enumext\* environments and the support for minipage and multicol environments using the keys columns, columns-sep, mini-env, mini-right and \miniright command. Source file  and tagged PDF .
  - The file enumext-03.tex contains the tests for the enumext and keyanspic environments activated by the save-ans key together with the save-sep and save-ref keys and the \printkeyans command. Source file  and tagged PDF .
  - The file enumext-04.tex contains the tests for the \anskey command and the anskey\* environment activated by the save-ans key along with the \getkeyans and \printkeyans commands. Source file  and tagged PDF .

- The file `enumext-05.tex` contains the tests for the environments `keyans`, `keyans*` and `keyanspic` activated by the key `save-ans` together with the keys `no-store` and `show-ans` and the commands `\setenumext`, `\setenumextmeta`, `\printkeyans` and `\foreachkeyans`. Source file  and tagged PDF .
- The file `enumext-06.tex` contains the tests for the environments `enumext` and `enumext*` for *fake itemize* and *description*. Source file  and tagged PDF .

## 9 The way of non-enumerated lists

It is possible to use (or abuse) the `enumext` and `enumext*` environments to mimic *non-enumerated* list environments such as `itemize` and `description`, clearly the `\keys` to “store answers”, the `keyans`, `keyans*` and `keyanspic` environments lose their sense and it is not the focus of `enumext` package, but, why not to do it?.

Here I leave as an example other uses of the `enumext` environment that can be helpful for specific purposes. The *trick* to generate these “fake environments” is set `label={}` or `label={\some}` and play with the `list-indent`, `list-offset`, `font` and `wrap-label` keys.

### Fake itemize environment

Here we set the `label` key using the default settings in  $\TeX$  for the four levels `\textbullet`, `\textendash`, `\textasteriskcentered` and `\textperiodcentered` together with the `nosep` key to reduce the vertical spaces in the left side example and set the `label` key in *mathematical mode* for the right side as `\ast`, `\diamond`, `\circ` and `\star` for the four levels together with the `nosep` key

- |                     |                     |
|---------------------|---------------------|
| • First level item  | * First level item  |
| – Second level item | ◇ Second level item |
| • Third level item  | ◦ Third level item  |
| · Fourth level item | ★ Fourth level item |
| • First level item  | * First level item  |

### Fake description environment

Here we set `label={}` and `list-indent=2.5em`, `font=\bfseries`.

**Something** A short one-line description.

This is an entry *without* a label.

**Something** A short *one-line* description text.

**Something long** A much *longer* description text may take more than one line or more than one paragraph.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

If we add `list-indent=0pt` you get *widest style*:

**Something** A short one-line description.

This is an entry *without* a label.

**Something** A short *one-line* description text.

**Something long** A much *longer* description text may take more than one line or more than one paragraph.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

🔗 The small space at the beginning of the “unlabeled entry” corresponds to `\labelsep` and can be removed using `\hspace{-\labelsep}` at the beginning of the line.

🔗 When *tagged PDF* is active the default `description` style is NOT available due to the redefinition of `\makelabel` for the `align` key which uses `\makebox` in this case, meaning that `\item[\langle content \rangle]` will not extend beyond `\labelwidth` which causes overlaps,

### Description indented by label

Here we set `label={}` and we will give a convenient value to `labelsep` and `labelwidth`, for example we can take as reference our *longest label* and pass it as value using:

```
\newlength{\descitemwd}
\settowidth{\descitemwd}{\textbf{Something long}}
```

and then use `labelsep=4pt`, `labelwidth=\descitemwd`, `font=\bfseries`.

**Something** A short one-line description.

This is an entry *without* a label.

**Something** A short one-line description.

**Something long** A much longer description. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris.

The environment can be translated so that the `\labels` are on the left margin calculating the value passed to the `list-offset` key, in this case it will be equal to the sum of the values set by the `labelwidth` and `labelsep` keys finally resulting as `list-offset={-\descitemwd - 4pt}`.

**Something**

A short one-line description.

|                |                                                                                                                                                                                          |
|----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                | This is an entry <i>without</i> a label.                                                                                                                                                 |
| Something      | A short one-line description.                                                                                                                                                            |
| Something long | A much longer description. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. |
|                | If we add <code>align=right</code> it will look like this:                                                                                                                               |
| Something      | A short one-line description.                                                                                                                                                            |
|                | This is an entry <i>without</i> a label.                                                                                                                                                 |
| Something      | A short one-line description.                                                                                                                                                            |
| Something long | A much longer description. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. |

At this point we have used `list-offset={-\descitemwd - 4pt}` instead of `list-offset={-\labelwidth - \labelsep}`, this is because the parameters `\labelwidth` and `\labelsep` take the default values, as if we had not set `label`.

Description with multi-line labels

The `label` key does not accept *multiline material*, this is where the `wrap-label` and `wrap-label*` keys comes into play. Unlike the `enumitem` package, the `align` key only supports three options, so what we will do is create a command in the style `\parleft` of `enumitem` that allows us to place *multiline labels* using `\parbox`.

```
\NewDocumentCommand \labelbx { s +m }
{%
 \SuspendTagging{\parbox}%
 \IfBooleanTF{#1}
 {%\strut\smash{\parbox[t]{\labelwidth}{\raggedright{#2}}}}%
 {%\strut\smash{\parbox[t]{\labelwidth}{\raggedleft{#2}}}}%
 \ResumeTagging{\parbox}%
}
```

Now we just need to set `wrap-label*={\labelbx{#1}}`.

|                |                                                                                                                                                                                          |
|----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Something      | A short one-line description.                                                                                                                                                            |
|                | This is an entry <i>without</i> a label.                                                                                                                                                 |
| Something      | A short one-line description.                                                                                                                                                            |
| Something long | A much longer description. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. |
|                | Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris.                            |
| SoMeThInG      | A much longer description. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. |
| LoNg           | ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris.                                                                                                               |

Final notes

The original implementation (if you can call it that) of the ideas that led to the creation of `enumext` were some macros using the `enumerate[5]` package for personal use created in early 2003, the code was quite questionable, but functional for these simple requirements.

With the great answers given by Christian Hupfer in [Create a fake label ref using list](#) and the answer given by David Carlisle in [Change the use of label ref by data save in an array \(list\)](#) I managed to create a more solid code than the original version, now using the `l3prop[11]` and `l3seq[11]` modules together with the `hyperref[8]` and `enumitem[6]` packages, which did the job, but with some limitations.

As time went by I took these limitations as a personal challenge which I called “*reinventing the wheel*”, since there were packages and classes that did more or less what I was looking for, but did not fit my simple requirements. This “*reinventing the wheel*” finally ended up becoming `enumext`.

Why list environments?

The answer is simple, first I love the beauty of its syntax and many of what I had already written used the `enumerate` environment or lists created using the `enumitem` package. In my mind I thought: how complicated could it be to write a package that looked like `enumitem`? It seemed simple enough, of course I didn’t have in mind the mess I was getting into working with `list` environments, `minipage` and adding support for the `multicol` and `hyperref` packages.

Of course, seeing the final result of the experiment “*reinventing the wheel*” I am quite satisfied.

Why not random questions and other utilities

The “*random*” type questions I love and hate them at the same time, although they simplify a lot the work when creating a multiple choice test, but you lose the beauty of typesetting a document with  $\text{\LaTeX}$ , that is to say the output does not always look as nice as it should, even if they are only alternatives these must follow a certain order when presented either numerical or presentation, that said handling that using *nested lists* is quite complicated so I do not classify to be implemented.



## Why has it taken so long?

One of the setbacks, beyond my laziness, was including compatibility with *tagged* PDF. To be honest, it's something I never considered at any point, but I firmly believe that being able to create *accessible documents* provides a great opportunity in the world of mathematics education. From my perspective as a *high school* teacher, beyond theorems and deep mathematics, the use of exercise lists is one of the most common things. Being able to open the way to work in parallel with those who have different abilities is really important and I regret not having looked into this in the past. I hope that `enumext` serves this purpose and inspires more users and authors to follow this path.

## 10 References

- [1] HIRSCHHORN, PHILIP. “Using the exam document class”. Available from CTAN, <https://www.ctan.org/pkg/exam>, 2023.
- [2] NIEDERBERGER, CLEMENS. “xsim – eXercise Sheets IMproved”. Available from CTAN, <https://www.ctan.org/pkg/xsim>, 2023.
- [3] MITTELBAACH, FRANK. “An environment for multicolumn output”. Available from CTAN, <https://www.ctan.org/pkg/multicol>, 2024.
- [4] GONZÁLEZ, PABLO. “scontents - Stores  $\LaTeX$  contents in memory or files”. Available from CTAN, <https://www.ctan.org/pkg/scontents>, 2024.
- [5] The  $\LaTeX$  Project. “enumerate – Enumerate with redefinable labels”. Available from CTAN, <https://www.ctan.org/pkg/enumerate>, 2024.
- [6] BEZOS, JAVIER. “Customizing lists with the enumitem package”. Available from CTAN, <https://www.ctan.org/pkg/enumitem>, 2019.
- [7] BERRY, KARL. “ $\LaTeX 2_{\epsilon}$ : An Unofficial Reference Manual”. Available from CTAN, <https://ctan.org/pkg/latex2e-help-texinfo>, 2024.
- [8] The  $\LaTeX$  Project. “Extensive support for hypertext in  $\LaTeX$ ”. Available from CTAN, <https://www.ctan.org/pkg/hyperref>, 2024.
- [9] BURNOL, JEAN-FRANÇOIS. “The footnotehyper package”. Available from CTAN, <https://www.ctan.org/pkg/footnotehyper>, 2021.
- [10] The  $\LaTeX$  Project. “The expl3 package”. Available from CTAN, <https://www.ctan.org/pkg/l3kernel>, 2024.
- [11] The  $\LaTeX$  Project. “The  $\LaTeX 3$  Interfaces”. Available from CTAN, <https://www.ctan.org/pkg/l3kernel>, 2024.
- [12] The  $\LaTeX$  Project. “The  $\LaTeX 2_{\epsilon}$  sources”. Available from CTAN, <https://ctan.org/tex-archive/macros/latex/base>, 2024.
- [13] The  $\LaTeX$  Project. “ $\LaTeX$  for authors current version”. Available from CTAN, <https://ctan.org/pkg/latex-base>, 2024.
- [14] GUNDLACH, PATRICK. “The lua-visual-debug package”. Available from CTAN, <https://www.ctan.org/pkg/lua-visual-debug>, 2023.
- [15] LEMVIG, MOGENS. “The shortlst package”. Available from CTAN, <https://www.ctan.org/pkg/shortlst>, 1998.
- [16] NIEDERBERGER, CLEMENS. “tasks – Horizontally columned lists”. Available from CTAN, <https://www.ctan.org/pkg/tasks>, 2022.
- [17] FISCHER, ULRIKE. “tagpdf –  $\LaTeX$  kernel code for PDF tagging”. Available from CTAN, <https://www.ctan.org/pkg/tagpdf>, 2024.
- [18] The  $\LaTeX$  Project. “ $\LaTeX$ -lab –  $\LaTeX$  laboratory”. Available from CTAN, <https://www.ctan.org/pkg/latex-lab>, 2024.
- [19] MITTELBAACH, FRANK. “ $\LaTeX$ ’s socket management”. Available from CTAN, <https://ctan.org/tex-archive/macros/latex/base>, 2024.

## 11 Change history

**v1.0 2024-11-03** – First public release.

12 Index of Documentation

The italic numbers denote the pages where the corresponding entry is described.

|                                  |                         |                                             |                               |
|----------------------------------|-------------------------|---------------------------------------------|-------------------------------|
| <b>C</b>                         |                         | <b>F</b>                                    |                               |
| Document class:                  |                         | \footnote                                   | 5                             |
| article                          | 2                       | <b>I</b>                                    |                               |
| book                             | 2                       | \itemsep                                    | 8                             |
| exam                             | 2                       | <b>K</b>                                    |                               |
| letter                           | 2                       | Keys for \anskey provide by enumext:        |                               |
| report                           | 2                       | break-col                                   | 13                            |
| \columnbreak                     | 4, 13                   | item-join                                   | 13                            |
| \columnsep                       | 10                      | item-pos*                                   | 13                            |
| Commands provide by enumext:     |                         | item-star                                   | 13                            |
| \anskey                          | 11-14                   | item-sym*                                   | 13                            |
| \anspic                          | 11, 12, 15, 16          | Keys for \foreachkeyans provide by enumext: |                               |
| \foreachkeyans                   | 17                      | after                                       | 18                            |
| \getkeyans                       | 12, 17                  | before                                      | 18                            |
| \item*                           | 5-7, 11, 12, 14, 15     | sep                                         | 17                            |
| \item                            | 5-7, 10-12, 14, 16      | start                                       | 17, 18                        |
| \miniright                       | 11                      | step                                        | 17                            |
| \printkeyans                     | 6, 12, 18               | stop                                        | 18                            |
| \setenumextmeta                  | 6                       | wrapper                                     | 18                            |
| \setenumext                      | 5-7, 11, 12, 15, 18     | Keys for anskey* provide by enumext:        |                               |
| Counters defined by enumext:     |                         | break-col                                   | 13                            |
| enumXiii                         | 4                       | force-eol                                   | 14                            |
| enumXii                          | 4                       | item-join                                   | 13                            |
| enumXiv                          | 4                       | item-pos*                                   | 13                            |
| enumXi                           | 4                       | item-star                                   | 13                            |
| enumXviii                        | 4                       | item-sym*                                   | 13                            |
| enumXvii                         | 4                       | overwrite                                   | 14                            |
| enumXvi                          | 4                       | write-env                                   | 14                            |
| enumXv                           | 4                       | Keys for environments provide by enumext:   |                               |
| <b>E</b>                         |                         | above*                                      | 9                             |
| Environments provide by enumext: |                         | above                                       | 8, 9                          |
| anskey*                          | 11-14, 21               | after                                       | 10                            |
| enumext*                         | 4-11, 13-15, 18, 21, 22 | align                                       | 7, 21-23                      |
| enumext                          | 4-11, 13-16, 18, 21, 22 | base-fix                                    | 8                             |
| keyans*                          | 4-14, 22                | before*                                     | 9, 10                         |
| keyanspic                        | 4, 7, 8, 11-16, 21, 22  | before                                      | 9                             |
| keyans                           | 4-16, 22                | below*                                      | 9                             |
| Environments:                    |                         | below                                       | 9                             |
| Verbatim                         | 14                      | check-ans                                   | 12, 13                        |
| center                           | 5                       | columns-sep                                 | 4, 10, 21                     |
| description                      | 5, 22                   | columns                                     | 4, 9, 10, 21                  |
| enumerate                        | 1, 3, 5, 23             | first                                       | 10                            |
| figure                           | 5                       | font                                        | 7                             |
| flushleft                        | 5                       | item-pos*                                   | 5, 6                          |
| flushright                       | 5                       | item-sym*                                   | 5, 6                          |
| itemize                          | 5, 22                   | itemindent                                  | 8-10                          |
| list                             | 3, 5, 9, 23             | itemsep                                     | 8                             |
| minipage                         | 3-5, 8-11, 21, 23       | label-pos                                   | 16                            |
| multicols                        | 3, 4, 10, 21            | label-sep                                   | 16                            |
| quotation                        | 5                       | labelsep                                    | 3-7, 9, 10, 12, 21, 22        |
| quote                            | 5                       | labelwidth                                  | 3, 4, 6, 7, 9, 10, 12, 21, 22 |
| shortenumberate                  | 5                       | labelwith                                   | 5                             |
| tabbing                          | 5                       | label                                       | 7, 8, 10, 15, 21-23           |
| table                            | 5                       | labewdith                                   | 9                             |
| tasks                            | 5                       | layout-sep                                  | 16                            |
| trivlist                         | 5                       | layout-sty                                  | 16                            |
| verbatim                         | 5                       | layout-top                                  | 16                            |
| verse                            | 5                       | list-indent                                 | 3, 9                          |
|                                  |                         | list-offset                                 | 3, 8, 9, 22, 23               |

|                            |                      |                |                      |
|----------------------------|----------------------|----------------|----------------------|
| listparindent              | 9, 10                | \Roman*        | 7, 8                 |
| mark-ans                   | 12                   | \alph*         | 7, 8                 |
| mark-pos                   | 12                   | \arabic*       | 7, 8                 |
| mark-ref                   | 12                   | \roman*        | 7, 8                 |
| mini-env                   | 4, 9, 11, 21         | \labelsep      | 3, 7                 |
| mini-right*                | 7, 11                | \labelwidth    | 3, 7                 |
| mini-right                 | 7, 11, 21            | \linewidth     | 11                   |
| mini-sep                   | 4, 11                | \listparindent | 9                    |
| mode-box                   | 7                    |                |                      |
| no-store                   | 11–13, 22            | <b>P</b>       |                      |
| noitemsep                  | 8                    | Packages:      |                      |
| nosep                      | 8, 22                | enumerate      | 23                   |
| overwrite                  | 14                   | enumext        | 1–5, 7, 16, 21–24    |
| parsep                     | 8, 10, 16            | enumitem       | 3, 4, 23             |
| partopsep                  | 8                    | fancyvrb       | 14                   |
| ref                        | 4, 8, 21             | footnotehyper  | 5                    |
| resume*                    | 7, 10, 11            | geometry       | 21                   |
| resume                     | 7, 10, 11            | graphicx       | 21                   |
| rightmargin                | 9                    | hyperref       | 4, 5, 12, 13, 21, 23 |
| save-ans                   | 4, 6, 10–18, 21, 22  | l3keys         | 7                    |
| save-key                   | 10–12, 18            | l3prop         | 23                   |
| save-ref                   | 4, 7, 12, 13, 17, 21 | l3seq          | 23                   |
| save-sep                   | 12, 21               | luamml         | 21                   |
| series                     | 7, 10, 11            | multicol       | 1, 2, 4, 21, 23      |
| show-ans                   | 12, 22               | scontents      | 1, 2, 14, 21         |
| show-length                | 8                    | shortlst       | 5                    |
| show-pos                   | 12, 17               | tasks          | 5                    |
| start*                     | 10                   | task           | 6                    |
| start                      | 10                   | unicode-math   | 21                   |
| topsep                     | 8, 9, 16             | xsim           | 2                    |
| widest                     | 7                    | \parsep        | 8                    |
| wrap-ans                   | 12                   | \partopsep     | 8                    |
| wrap-label*                | 7, 23                |                |                      |
| wrap-label                 | 7, 21, 23            | <b>R</b>       |                      |
| wrap-opt                   | 12                   | \raggedcolumns | 4                    |
| write-env                  | 14                   | \ref           | 4                    |
|                            |                      | \rightmargin   | 9                    |
| <b>L</b>                   |                      |                |                      |
| \label                     | 4                    | <b>T</b>       |                      |
| Labels provide by enumext: |                      | \topsep        | 8                    |
| \Alph*                     | 7, 8, 15             |                |                      |

## 13 Implementation

The most recent publicly released version of `enumext` is available at CTAN: <https://www.ctan.org/pkg/enumext>. While general feedback via email is welcomed, specific bugs or feature requests should be reported through the issue tracker: <https://github.com/pablgonz/enumext/issues>.

- The documentation presented here is far from professional, it contains a lot of obvious information that to the eye of a TeXpert are superfluous, but, after so many years developing this project is the only way to remember what does what.

### 13.1 General conventions

Variables containing `i`, `ii`, `iii` and `iv` are associated by level with the `enumext` environment, variables containing `v` are associated with the `keyans` environment, variables containing `vi` are associated with the `keyanspic` environment, variables containing `vii` are associated with the `enumext*` environment and variables containing `viii` are associated with the `keyans*` environment.

To simplify writing and documentation some variables and functions that are common to the different levels of the environments are described using a capital “X”.

The temporary function `\__enumext_tmp:n` is used in different parts of the package code for variable creation or execution of other functions that are grouped into this one.

All variables and functions defined in this package are private and are NOT intended to work or be used by another package or module.

### 13.2 Initial set up

Start the DocStrip guards.

```
1 <{*package>
```

Identify the internal prefix (L<sup>A</sup>T<sub>E</sub>X3 DocStrip convention) for l3doc class.

```
2 <@@=enumext>
```

### 13.3 Declaration of the package

First we will make sure we have a minimum (super updated) version of L<sup>A</sup>T<sub>E</sub>X to work correctly.

```
3 \NeedsTeXFormat{LaTeX2e}[2024-11-01]
```

Now declare the `enumext` package.

```
4 \ProvidesExplPackage {enumext} {2024-11-03} {1.0} {Enumerate exercise sheets}
```

Finally check if the `multicol` and `scontents` packages are loaded, if not we load it.

```
5 \hook_gput_code:nnn {begindocument} {enumext}
6 {
7 \IfPackageLoadedTF { multicol }
8 {
9 \msg_info:nnn { enumext } { package-load } { multicol }
10 }
11 {
12 \msg_info:nnn { enumext } { package-not-load } { multicol }
13 \RequirePackage{multicol}[2024-05-23]
14 }
15 \IfPackageLoadedTF { scontents }
16 {
17 \msg_info:nnn { enumext } { package-load } { scontents }
18 }
19 {
20 \msg_info:nnn { enumext } { package-not-load } { scontents }
21 \RequirePackage{scontents}
22 }
23 }
```

### 13.4 Definition of variables

Variables that do not appear in this section are created by means of `\keys_define:nn` or some function described below.

```
__enumext_level_int
__enumext_level_h_int
__enumext_anskey_level_int
__enumext_keyans_level_int
__enumext_keyans_level_h_int
__enumext_keyans_pic_level_int

24 \int_new:N __enumext_level_int
25 \int_new:N __enumext_level_h_int
26 \int_new:N __enumext_anskey_level_int
27 \int_new:N __enumext_keyans_level_int
28 \int_new:N __enumext_keyans_level_h_int
29 \int_new:N __enumext_keyans_pic_level_int
```

Integer variables will control the nesting levels of the environments and `\anskey` command.

(End of definition for `\__enumext_level_int` and others.)

```

\l__enumext_starred_bool
\g__enumext_starred_bool
\l__enumext_starred_first_bool
\l__enumext_standar_bool
\g__enumext_standar_bool
\l__enumext_standar_first_bool
\l__enumext_anskey_env_bool
\l__enumext_keyans_env_bool
\g__enumext_start_line_tl
\g__enumext_envir_name_tl
\l__enumext_envir_name_tl

```

Internal variables used by functions `\__enumext_is_not_nested:`, `\__enumext_is_on_first_level:` and `\__enumext_keyans_name_and_start:` (§13.5.1).

```

30 \bool_new:N \l__enumext_starred_bool
31 \bool_new:N \g__enumext_starred_bool
32 \bool_new:N \l__enumext_starred_first_bool
33 \bool_new:N \l__enumext_standar_bool
34 \bool_new:N \g__enumext_standar_bool
35 \bool_new:N \l__enumext_standar_first_bool
36 \bool_new:N \l__enumext_anskey_env_bool
37 \bool_new:N \l__enumext_keyans_env_bool
38 \tl_new:N \g__enumext_start_line_tl
39 \tl_new:N \g__enumext_envir_name_tl
40 \tl_new:N \l__enumext_envir_name_tl

```

(End of definition for `\l__enumext_starred_bool` and others.)

```

\l__enumext_counter_i_tl
\l__enumext_counter_ii_tl
\l__enumext_counter_iii_tl
\l__enumext_counter_iv_tl
\l__enumext_counter_v_tl
\l__enumext_counter_vi_tl
\l__enumext_counter_vii_tl
\l__enumext_counter_viii_tl

```

Variables to store the “*name of the counters*” `enumXi`, `enumXii`, `enumXiii` and `enumXiv` for `enumext` environment, `enumXv` for `keyans` environment and `enumXvi` for the `keyanspic` environment. The counters `enumXvii` and `enumXviii` are used by `enumext*` and `keyans*` environments.

The initial values of these variables are set by the function `\__enumext_define_counters:Nn` (§13.11) and then modified by the function `\__enumext_label_style:Nnn` used by `label` key (§13.14).

```

41 \cs_set_protected:Npn __enumext_tmp:n #1
42 {
43 \tl_new:c { l__enumext_counter_#1_tl }
44 }
45 \clist_map_inline:nn { i, ii, iii, iv, v, vi, vii, viii } { __enumext_tmp:n {#1} }

```

(End of definition for `\l__enumext_counter_i_tl` and others.)

```

\c__enumext_counter_style_tl
\l__enumext_ref_key_arg_tl
\l__enumext_ref_the_count_tl
\l__enumext_the_counter_X_tl
\l__enumext_renew_the_count_X_tl

```

Internal variables used by `ref` key (§13.14).

```

46 \tl_const:Nn \c__enumext_counter_style_tl
47 { { { arabic } { roman } { Roman } { alph } { Alph } } }
48 \tl_new:N \l__enumext_ref_key_arg_tl
49 \tl_new:N \l__enumext_ref_the_count_tl
50 \cs_set_protected:Npn __enumext_tmp:n #1
51 {
52 \tl_new:c { l__enumext_renew_the_count_#1_tl }
53 \tl_new:c { l__enumext_the_counter_#1_tl }
54 \tl_set:ce { l__enumext_the_counter_#1_tl } { \exp_not:c { theenumX#1 } }
55 }
56 \clist_map_inline:nn { i, ii, iii, iv, v, vi, vii, viii } { __enumext_tmp:n {#1} }

```

(End of definition for `\c__enumext_counter_style_tl` and others.)

```

\g__enumext_resume_int
\g__enumext_resume_vii_int
\l__enumext_resume_name_tl
\l__enumext_resume_active_bool
\g__enumext_starred_series_tl
\g__enumext_standar_series_tl

```

Internal variables used by `resume`, `resume*` and `series` keys (§13.25).

```

57 \int_new:N \g__enumext_resume_int
58 \int_new:N \g__enumext_resume_vii_int
59 \tl_new:N \l__enumext_resume_name_tl
60 \bool_new:N \l__enumext_resume_active_bool
61 \tl_new:N \g__enumext_standar_series_tl
62 \tl_new:N \g__enumext_starred_series_tl

```

(End of definition for `\g__enumext_resume_int` and others.)

```

\l__enumext_current_widest_dim
\g__enumext_counter_styles_tl
\g__enumext_widest_label_tl
\l__enumext_label_width_by_box

```

The variable `\l__enumext_current_widest_dim` stores the current label width, the variable `\g__enumext_counter_styles_tl` stores the default *label style* and the variable `\g__enumext_widest_label_tl` the label width. These variables are used by `widest` (§13.15) and `label` (§13.13) keys.

```

63 \dim_new:N \l__enumext_current_widest_dim
64 \tl_new:N \g__enumext_counter_styles_tl
65 \tl_new:N \g__enumext_widest_label_tl
66 \box_new:N \l__enumext_label_width_by_box

```

(End of definition for `\l__enumext_current_widest_dim` and others.)



```
\l__enumext_leftmargin_tmp_X_bool
\l__enumext_leftmargin_tmp_X_dim
\l__enumext_leftmargin_X_dim
\l__enumext_itemindent_X_dim
```

The boolean variable `\l__enumext_leftmargin_tmp_X_bool` and the dimensional variable `\l__enumext_leftmargin_tmp_X_dim` are used by the `list-indent` key (§13.18). The variables `\l__enumext_leftmargin_X_dim` and `\l__enumext_itemindent_X_dim` are used and set by the function `\__enumext_calc_hspace:NNNNNNNNNN` (§13.38.1).

```
67 \cs_set_protected:Npn __enumext_tmp:n #1
68 {
69 \bool_new:c { \l__enumext_leftmargin_tmp_#1_bool }
70 \dim_new:c { \l__enumext_leftmargin_tmp_#1_dim }
71 \dim_new:c { \l__enumext_leftmargin_#1_dim }
72 \dim_new:c { \l__enumext_itemindent_#1_dim }
73 }
74 \clist_map_inline:nn { i, ii, iii, iv, v, vi, vii, viii } { __enumext_tmp:n {#1} }
```

(End of definition for `\l__enumext_leftmargin_tmp_X_bool` and others.)

```
\l__enumext_multicols_above_X_skip
\l__enumext_multicols_below_X_skip
\g__enumext_multicols_right_X_skip
\l__enumext_align_label_pos_X_str
```

Internal variables used by `columns` key (§13.22) and `align` key (§13.13).

```
75 \cs_set_protected:Npn __enumext_tmp:n #1
76 {
77 \skip_new:c { \l__enumext_multicols_above_#1_skip }
78 \skip_new:c { \l__enumext_multicols_below_#1_skip }
79 \skip_new:c { \g__enumext_multicols_right_#1_skip }
80 \str_new:c { \l__enumext_align_label_pos_#1_str }
81 }
82 \clist_map_inline:nn { i, ii, iii, iv, v } { __enumext_tmp:n {#1} }
```

(End of definition for `\l__enumext_multicols_above_X_skip` and others.)

```
\g__enumext_minipage_stat_int
\l__enumext_minipage_temp_skip
\l__enumext_minipage_left_skip
\l__enumext_minipage_right_skip
\l__enumext_minipage_after_skip
\g__enumext_minipage_right_skip
\g__enumext_minipage_after_skip
\l__enumext_minipage_left_X_dim
\l__enumext_minipage_active_X_bool
```

Internal variables used by `\miniright` command (§13.23.4) and the keys `mini-right`, `mini-right*`, `mini-env` and `mini-sep` (§13.21, §13.23).

```
83 \int_new:N \g__enumext_minipage_stat_int
84 \skip_new:N \l__enumext_minipage_temp_skip
85 \skip_new:N \l__enumext_minipage_left_skip
86 \skip_new:N \l__enumext_minipage_right_skip
87 \skip_new:N \l__enumext_minipage_after_skip
88 \skip_new:N \g__enumext_minipage_right_skip
89 \skip_new:N \g__enumext_minipage_after_skip
90 \cs_set_protected:Npn __enumext_tmp:n #1
91 {
92 \dim_new:c { \l__enumext_minipage_left_#1_dim }
93 \bool_new:c { \l__enumext_minipage_active_#1_bool }
94 }
95 \clist_map_inline:nn { i, ii, iii, iv, v, vii, viii } { __enumext_tmp:n {#1} }
```

(End of definition for `\g__enumext_minipage_stat_int` and others.)

```
\l__enumext_wrap_label_X_bool
\l__enumext_wrap_label_opt_X_bool
\l__enumext_start_X_int
\l__enumext_fake_item_indent_X_tl
\l__enumext_label_fill_left_X_tl
\l__enumext_label_fill_right_X_tl
\l__enumext_vspace_a_star_X_bool
\l__enumext_vspace_b_star_X_bool
```

The bool vars `\l__enumext_wrap_label_X_bool` and `\l__enumext_wrap_label_opt_X_bool` are used by `wrap-label` and `wrap-label*` keys (§13.13), the integer `\l__enumext_start_X_int` are used by the `start` and `start*` keys (§13.15), the token list `\l__enumext_fake_item_indent_X_tl` is used by `itemindent` key (§13.18.1), the variables `\l__enumext_label_fill_left_X_tl` and `\l__enumext_label_fill_right_X_tl` are used by the `align` key (§13.13). The boolean vars `\l__enumext_vspace_a_star_X_bool`, `\l__enumext_vspace_b_star_X_bool` are used by `above`, `above*`, `below` and `below*` keys (§13.20).

```
96 \cs_set_protected:Npn __enumext_tmp:n #1
97 {
98 \bool_new:c { \l__enumext_wrap_label_#1_bool }
99 \bool_new:c { \l__enumext_wrap_label_opt_#1_bool }
100 \int_new:c { \l__enumext_start_#1_int }
101 \tl_new:c { \l__enumext_fake_item_indent_#1_tl }
102 \tl_new:c { \l__enumext_label_fill_left_#1_tl }
103 \tl_new:c { \l__enumext_label_fill_right_#1_tl }
104 \bool_new:c { \l__enumext_vspace_a_star_#1_bool }
105 \bool_new:c { \l__enumext_vspace_b_star_#1_bool }
106 }
107 \clist_map_inline:nn { i, ii, iii, iv, v, vii, viii } { __enumext_tmp:n {#1} }
```

(End of definition for `\l__enumext_wrap_label_X_bool` and others.)

```

\l__enumext_store_active_bool
\l__enumext_store_name_tl
\g__enumext_store_name_tl
\l__enumext_store_anskey_arg_tl
\l__enumext_store_anskey_env_tl
\l__enumext_store_anskey_opt_tl
\l__enumext_store_current_label_tl
\l__enumext_store_current_opt_arg_tl
\l__enumext_store_current_label_tmp_tl

```

The variable `\l__enumext_store_active_bool` setting by `save-ans` key (§13.26.1) activates all the mechanism related to `\anskey`, `anskey*`, `keyans`, `keyans*` and `keyanspic` environments.

The variable `\l__enumext_store_name_tl` saves the  $\{\langle store\ name\rangle\}$  set by the `save-ans` key of the *sequence* and *prop list* in which we will store, the variable `\g__enumext_store_name_tl` it's just a global copy of  $\{\langle store\ name\rangle\}$  used by different functions.

The variable `\l__enumext_store_anskey_arg_tl` save the *argument* of `\anskey` (§13.30) and the variables `\l__enumext_store_anskey_env_tl` and `\l__enumext_store_anskey_opt_tl` save the  $\langle body\rangle$  and the  $\langle keys\rangle$  of the environment `anskey*` (§13.31).

The variables `\l__enumext_store_current_label_tl` and `\l__enumext_store_current_opt_arg_tl` save the *current label* and *optional argument* of `\item*` (§13.37) and `\anspic*` (§13.42.2) for the `keyans`, `keyans*` and `keyanspic` environments.

The variable `\l__enumext_store_current_label_tmp_tl` is a temporary variable used by `keyans`, `keyans*` and `keyanspic` at various points.

```

108 \bool_new:N \l__enumext_store_active_bool
109 \tl_new:N \l__enumext_store_name_tl
110 \tl_new:N \g__enumext_store_name_tl
111 \tl_new:N \l__enumext_store_anskey_arg_tl
112 \tl_new:N \l__enumext_store_anskey_env_tl
113 \tl_new:N \l__enumext_store_anskey_opt_tl
114 \tl_new:N \l__enumext_store_current_label_tl
115 \tl_new:N \l__enumext_store_current_opt_arg_tl
116 \tl_new:N \l__enumext_store_current_label_tmp_tl

```

(End of definition for `\l__enumext_store_active_bool` and others.)

```

\l__enumext_setkey_tmpa_tl
\l__enumext_setkey_tmpb_tl
\l__enumext_setkey_tmpa_int
\l__enumext_setkey_tmpa_seq
\l__enumext_setkey_tmpb_seq

```

Internal variables used by the command `\setenumext` (§13.48).

```

117 \tl_new:N \l__enumext_setkey_tmpa_tl
118 \tl_new:N \l__enumext_setkey_tmpb_tl
119 \int_new:N \l__enumext_setkey_tmpa_int
120 \seq_new:N \l__enumext_setkey_tmpa_seq
121 \seq_new:N \l__enumext_setkey_tmpb_seq

```

(End of definition for `\l__enumext_setkey_tmpa_tl` and others.)

```

\l__enumext_meta_path_tl
\l__enumext_foreach_print_seq
\l__enumext_foreach_name_prop_tl
\g__enumext_foreach_default_keys_tl

```

Internal variables used by the `\printkeyans` command (§13.47) and `\foreachkeyans` command (§13.50).

```

122 \tl_new:N \l__enumext_meta_path_tl
123 \seq_new:N \l__enumext_foreach_print_seq
124 \tl_new:N \l__enumext_foreach_name_prop_tl
125 \tl_new:N \g__enumext_foreach_default_keys_tl

```

(End of definition for `\l__enumext_meta_path_tl` and others.)

```

\l__enumext_print_keyans_starred_tl
\l__enumext_print_keyans_star_bool
\l__enumext_mark_position_str
\g__enumext_item_symbol_aux_tl
\l__enumext_print_keyans_X_tl
\l__enumext_store_save_key_X_tl
\l__enumext_store_save_key_X_bool
\l__enumext_store_upper_level_X_bool

```

Internal variables used by command `\printkeyans` (§13.47), `show-pos` key (§13.27), `item-sym*` key (§13.35), `save-key` key (§13.27.2) and “*storing structure*”.

```

126 \tl_new:N \l__enumext_print_keyans_starred_tl
127 \bool_new:N \l__enumext_print_keyans_star_bool
128 \str_new:N \l__enumext_mark_position_str
129 \tl_new:N \g__enumext_item_symbol_aux_tl
130 \cs_set_protected:Npn __enumext_tmp:n #1
131 {
132 \tl_new:c { \l__enumext_print_keyans_#1_tl }
133 \tl_new:c { \l__enumext_store_save_key_#1_tl }
134 \bool_new:c { \l__enumext_store_save_key_#1_bool }
135 \bool_new:c { \l__enumext_store_upper_level_#1_bool }
136 }
137 \clist_map_inline:nn { i, ii, iii, iv, vii } { __enumext_tmp:n {#1} }

```

(End of definition for `\l__enumext_print_keyans_starred_tl` and others.)

```

\l__enumext_anspic_args_seq
\l__enumext_anspic_mini_width_dim
\l__enumext_anspic_above_int
\l__enumext_anspic_below_int
\l__enumext_anspic_label_above_bool
\l__enumext_anspic_mini_pos_str
\g__enumext_keyans_pic_parsep_skip
\l__enumext_anspic_label_box
\l__enumext_anspic_body_box
\l__enumext_anspic_label_htdp_dim
\l__enumext_anspic_body_htdp_dim

```

Internal variables used by `keyanspic` environment and `\anspic` command (§13.42.1).

```

138 \seq_new:N \l__enumext_anspic_args_seq
139 \dim_new:N \l__enumext_anspic_mini_width_dim
140 \int_new:N \l__enumext_anspic_above_int
141 \int_new:N \l__enumext_anspic_below_int
142 \bool_new:N \l__enumext_anspic_label_above_bool
143 \str_new:N \l__enumext_anspic_mini_pos_str
144 \skip_new:N \g__enumext_keyans_pic_parsep_skip
145 \box_new:N \l__enumext_anspic_label_box
146 \box_new:N \l__enumext_anspic_body_box
147 \dim_new:N \l__enumext_anspic_label_htdp_dim
148 \dim_new:N \l__enumext_anspic_body_htdp_dim

```

(End of definition for `\l__enumext_anspic_args_seq` and others.)

```
\l__enumext_check_answers_bool
\g__enumext_check_ans_key_bool
\l__enumext_check_start_line_env_tl
\l__enumext_item_star_exec_int
\l__enumext_item_star_wrap_bool
\g__enumext_check_starred_cmd_int
\g__enumext_item_anskey_int
\g__enumext_item_number_int
\g__enumext_item_number_bool
\g__enumext_item_answer_diff_int
```

Internal variables used by “*internal check answer*” mechanism (§13.26.3) used by the `check-ans`, `no-store`, `wrap-key` keys and check for starred commands `\item*` in `keyans` and `keyans*` environments and `\anspic*` in `keyanspic` environment.

```
149 \bool_new:N \l__enumext_check_answers_bool
150 \bool_new:N \g__enumext_check_ans_key_bool
151 \tl_new:N \l__enumext_check_start_line_env_tl
152 \int_new:N \l__enumext_item_star_exec_int
153 \bool_new:N \l__enumext_item_star_wrap_bool
154 \int_new:N \g__enumext_check_starred_cmd_int
155 \int_new:N \g__enumext_item_anskey_int
156 \int_new:N \g__enumext_item_number_int
157 \bool_new:N \l__enumext_item_number_bool
158 \int_new:N \g__enumext_item_answer_diff_int
```

(End of definition for `\l__enumext_check_answers_bool` and others.)

```
\l__enumext_hyperref_bool
\l__enumext_footnotes_key_bool
```

The boolean variable `\l__enumext_hyperref_bool` will determine if the `hyperref` package is present or load in memory (§13.7). The boolean variable `\l__enumext_footnotes_key_bool` determine if `hyperref` is load with key `hyperfootnotes=true`.

```
159 \bool_new:N \l__enumext_hyperref_bool
160 \bool_new:N \l__enumext_footnotes_key_bool
```

(End of definition for `\l__enumext_hyperref_bool` and `\l__enumext_footnotes_key_bool`.)

```
\l__enumext_newlabel_arg_one_tl
\l__enumext_newlabel_arg_two_tl
\l__enumext_write_aux_file_tl
\l__enumext_label_copy_X_tl
```

Internal variables used by `save-ref` key (§13.27). The variables `\l__enumext_label_copy_X_tl` correspond to temporary copies of the  $\langle labels \rangle$  defined by level on which operations will be performed.

The variables `\l__enumext_newlabel_arg_one_tl` and `\l__enumext_newlabel_arg_two_tl` will be used to form the arguments passed to the function `\__enumext_newlabel:nn` (§13.7) and the variable `\l__enumext_write_aux_file_tl` will be in charge of executing the writing code in the `.aux` file.

```
161 \tl_new:N \l__enumext_newlabel_arg_one_tl
162 \tl_new:N \l__enumext_newlabel_arg_two_tl
163 \tl_new:N \l__enumext_write_aux_file_tl
164 \cs_set_protected:Npn __enumext_tmp:n #1
165 {
166 \tl_new:c { l__enumext_label_copy_#1_tl }
167 }
168 \clist_map_inline:nn { i, ii, iii, iv, v, vi, vii, viii } { __enumext_tmp:n {#1} }
```

(End of definition for `\l__enumext_newlabel_arg_one_tl` and others.)

```
\g__enumext_footnote_standar_int
\g__enumext_footnote_starred_int
\g__enumext_footnote_standar_arg_seq
\g__enumext_footnote_starred_arg_seq
\g__enumext_footnote_standar_int_seq
\g__enumext_footnote_starred_int_seq
```

Internal variables used for redefinition of `\footnote` (§13.8).

```
169 \int_new:N \g__enumext_footnote_standar_int
170 \int_new:N \g__enumext_footnote_starred_int
171 \seq_new:N \g__enumext_footnote_standar_arg_seq
172 \seq_new:N \g__enumext_footnote_starred_arg_seq
173 \seq_new:N \g__enumext_footnote_standar_int_seq
174 \seq_new:N \g__enumext_footnote_starred_int_seq
```

(End of definition for `\g__enumext_footnote_standar_int` and others.)

```
\l__enumext_item_starred_X_bool
\l__enumext_item_column_pos_X_int
\g__enumext_item_count_all_X_int
\l__enumext_joined_item_X_int
\l__enumext_joined_item_aux_X_int
\l__enumext_tmpa_X_int
\l__enumext_tmpa_X_dim
\l__enumext_item_text_X_box
\l__enumext_joined_width_X_dim
\l__enumext_item_width_X_dim
\g__enumext_item_symbol_aux_X_tl
\l__enumext_align_label_X_str
\g__enumext_minipage_active_X_bool
\l__enumext_miniright_code_X_box
\g__enumext_minipage_center_X_bool
\g__enumext_minipage_right_X_dim
\g__enumext_minipage_right_X_skip
```

Internal variables used by `enumext*` and `keyans*` environments.

```
175 \cs_set_protected:Npn __enumext_tmp:n #1
176 {
177 \bool_new:c { l__enumext_item_starred_#1_bool }
178 \int_new:c { l__enumext_item_column_pos_#1_int }
179 \int_new:c { g__enumext_item_count_all_#1_int }
180 \int_new:c { l__enumext_joined_item_#1_int }
181 \int_new:c { l__enumext_joined_item_aux_#1_int }
182 \int_new:c { l__enumext_tmpa_#1_int }
183 \dim_new:c { l__enumext_tmpa_#1_dim }
184 \box_new:c { l__enumext_item_text_#1_box }
185 \dim_new:c { l__enumext_joined_width_#1_dim }
186 \dim_new:c { l__enumext_item_width_#1_dim }
187 \tl_new:c { g__enumext_item_symbol_aux_#1_tl }
188 \str_new:c { l__enumext_align_label_#1_str }
189 \bool_new:c { g__enumext_minipage_active_#1_bool }
190 \box_new:c { l__enumext_miniright_code_#1_box }
191 \bool_new:c { g__enumext_minipage_center_#1_bool }
```

```

192 \dim_new:c { g__enumext_minipage_right_#1_dim }
193 \skip_new:c { g__enumext_minipage_right_#1_skip }
194 }
195 \clist_map_inline:nn { vii, viii } { __enumext_tmp:n {#1} }

```

(End of definition for `\__enumext_item_starred_X_bool` and others.)

`\c__enumext_all_envs_clist` An internal `clist-var` variable to run with `\__enumext_tmp:n`.

```

196 \clist_const:Nn \c__enumext_all_envs_clist
197 {
198 {level-1}{i}, {level-2}{ii}, {level-3}{iii}, {level-4}{iv},
199 {keyans}{v}, {enumext*}{vii}, {keyans*}{viii}
200 }

```

(End of definition for `\c__enumext_all_envs_clist`.)

### 13.5 Some utility functions

`\keys_precompile:neN` `\seq_use:NV` Non-standard kernel variants used by the `\printkeyans` command (§13.47) and `\foreachkeyans` command (§13.50).

```

201 \cs_generate_variant:Nn \keys_precompile:nnN { neN }
202 \cs_generate_variant:Nn \seq_use:Nn { NV }

```

(End of definition for `\keys_precompile:neN` and `\seq_use:NV`.)

`\__enumext_at_begin_document:n` A internal “hook” function used for copying plain `list` and `minipage` environments definition and `hyperref` detection.

```

203 \cs_new_protected:Npn __enumext_at_begin_document:n #1
204 {
205 \hook_gput_code:nnn {begindocument} {enumext} { #1 }
206 }

```

(End of definition for `\__enumext_at_begin_document:n`.)

`\__enumext_after_env:nn` `\__enumext_before_env:nn` A internal “hook” functions for execute code `mini-right` and `mini-right*` keys outside the `enumext*` and `keyans*` environments and print check-ans outside the `enumext` and `enumext*` environments.

```

207 \cs_new_protected:Npn __enumext_after_env:nn #1 #2
208 {
209 \hook_gput_code:nnn {env/#1/after} {enumext} {#2}
210 }
211 \cs_new_protected:Npn __enumext_before_env:nn #1 #2
212 {
213 \hook_gput_code:nnn {env/#1/before} {enumext} {#2}
214 }

```

(End of definition for `\__enumext_after_env:nn` and `\__enumext_before_env:nn`.)

`\__enumext_level:` Function for check current level in `enumext`.

```

215 \cs_new:Nn __enumext_level:
216 {
217 \int_to_roman:n { __enumext_level_int }
218 }

```

(End of definition for `\__enumext_level:`.)

`\__enumext_if_is_int:nT` `\__enumext_if_is_int:nF` `\__enumext_if_is_int:nTF` A conditional function to know if the variable we are passing is an integer used by `start` and `widest` keys. This function is taken directly from the answer given by Henri Menke in [How to test if an expl3 function argument is an integer expression?](#).

```

219 \prg_new_protected_conditional:Npnn __enumext_if_is_int:n #1 { T, F, TF }
220 {
221 \regex_match:nnTF { ^[\+|-]?[\d]+$ } {#1} % $
222 { \prg_return_true: }
223 { \prg_return_false: }
224 }

```

(End of definition for `\__enumext_if_is_int:nT`, `\__enumext_if_is_int:nF`, and `\__enumext_if_is_int:nTF`.)

`\__enumext_regex_counter_style:`

The internal function `\__enumext_regex_counter_style:` replace the ‘`*`’ with the actual counter of the running level and is used by the `ref` key. It loops through the defined counter styles in `\c__enumext_counter_style_tl` and replace ‘`*`’ by real command, for example, looking for `\arabic*` and replacing that by `\arabic{<counter>}` defined on the current level.

```

225 \cs_new_protected:Nn __enumext_regex_counter_style:
226 {
227 \tl_map_inline:Nn \c__enumext_counter_style_tl
228 {
229 \regex_replace_once:nnN { \c{##1}* }
230 { \c{##1}\cB{\u{__enumext_ref_the_count_tl}\cE} } \l__enumext_ref_key_arg_tl
231 }
232 }

```

(End of definition for `\__enumext_regex_counter_style:.`)

`\__enumext_show_length:nnn`

Internal function used by `show-length` key to show “all lengths” calculated and use in `enumext`, `enumext*`, `keyans` and `keyans*` environments.

```

233 \cs_new:Npn __enumext_show_length:nnn #1 #2 #3
234 {
235 * ~ #2
236 \prg_replicate:nn { 14 - \str_count:n {#2} } { ~ }
237 = ~ \use:c { #1_use:c } { l__enumext_#2_#3_#1 } \\
238 }

```

(End of definition for `\__enumext_show_length:nnn.`)

`\__enumext_unskip_unkern:`

The function `\__enumext_unskip_unkern:` will remove the last `<skip>` or `<kern>` at execution time using the values `11` and `12` of `\lastnodetype` to apply `\unskip` or `\unkern` according to the case.

```

239 \cs_new_protected:Nn __enumext_unskip_unkern:
240 {
241 \int_case:nnT { \lastnodetype }
242 {
243 { 11 }{ \unskip }
244 { 12 }{ \unkern }
245 }
246 }

```

(End of definition for `\__enumext_unskip_unkern:.`)

### 13.5.1 Utilities for environments and levels

`\__enumext_is_not_nested:``\__enumext_is_on_first_level:`

The function `\__enumext_is_not_nested:` set the variables `\g__enumext_standar_bool` and `\g__enumext_starred_bool` to “true” only if the environments `enumext` and `enumext*` are NOT nested in each other and save the environment name in `\l__enumext_envir_name_tl`.

```

247 \cs_new_protected:Nn __enumext_is_not_nested:
248 {
249 \str_case:en { \@currentenvir }
250 {
251 {enumext}
252 {
253 \tl_set:Nn \l__enumext_envir_name_tl { enumext }
254 \bool_lazy_and:nnT
255 { \bool_not_p:n { \g__enumext_standar_bool } }
256 { \int_compare_p:nNn { \l__enumext_level_h_int } = { 0 } }
257 {
258 \bool_gset_true:N \g__enumext_standar_bool
259 }
260 }
261 {enumext*}
262 {
263 \tl_set:Nn \l__enumext_envir_name_tl { enumext* }
264 \bool_lazy_and:nnT
265 { \bool_not_p:n { \g__enumext_starred_bool } }
266 { \int_compare_p:nNn { \l__enumext_level_int } = { 0 } }
267 {
268 \bool_gset_true:N \g__enumext_starred_bool
269 }
270 }
271 }
272 }

```

The function `\__enumext_is_on_first_level:` will set the variables `\l__enumext_standar_first_bool` (§13.26.1), `\l__enumext_starred_first_bool` (§13.26.1) and `\l__enumext_anskey_env_bool` (§13.31) to “true” only if the environment is not nested and we are in the “first level” of it . We will also save the *start line number* of each environment in the variable `\g__enumext_start_line_tl` and the *name* of each environment in the variable `\g__enumext_envir_name_tl` to use in messages related to the `check-ans` key and `.log` file.

```

273 \cs_new_protected:Nn __enumext_is_on_first_level:
274 {
275 \bool_lazy_all:nT
276 {
277 { \bool_if_p:N \g__enumext_standar_bool }
278 { \int_compare_p:nNn { \l__enumext_level_int } = { 1 } }
279 { \int_compare_p:nNn { \l__enumext_level_h_int } = { 0 } }
280 }
281 {
282 \bool_set_true:N \l__enumext_standar_first_bool
283 \bool_set_true:N \l__enumext_anskey_env_bool
284 \tl_gset:Nn \g__enumext_envir_name_tl { enumext }
285 \tl_gset:Ne \g__enumext_start_line_tl
286 {
287 on ~ line ~ \exp_not:V \inputlineno
288 }
289 }
290 \bool_lazy_all:nT
291 {
292 { \bool_if_p:N \g__enumext_starred_bool }
293 { \int_compare_p:nNn { \l__enumext_level_h_int } = { 1 } }
294 { \int_compare_p:nNn { \l__enumext_level_int } = { 0 } }
295 }
296 {
297 \bool_set_true:N \l__enumext_starred_first_bool
298 \bool_set_true:N \l__enumext_anskey_env_bool
299 \tl_gset:Nn \g__enumext_envir_name_tl { enumext* }
300 \tl_gset:Ne \g__enumext_start_line_tl
301 {
302 on ~ line ~ \exp_not:V \inputlineno
303 }
304 }
305 }

```

(End of definition for `\__enumext_is_not_nested:` and `\__enumext_is_on_first_level:`)

`\__enumext_keyans_name_and_start:`

The function `\__enumext_keyans_name_and_start:` will save the start line number and name of the environments `keyans`, `keyans*` and `keyanspic` in the variables `\l__enumext_check_start_line_env_tl` and `\l__enumext_envir_name_tl` to use in the `\__enumext_check_starred_cmd:n` function.

```

306 \cs_new_protected:Nn __enumext_keyans_name_and_start:
307 {
308 \str_case:en { \@currenvir }
309 {
310 {keyans}
311 {
312 \tl_set:Nn \l__enumext_envir_name_tl { keyans }
313 \tl_set:Ne \l__enumext_check_start_line_env_tl
314 {
315 in ~ 'keyans' ~ start ~ on ~ line ~ \exp_not:V \inputlineno
316 }
317 }
318 {keyans*}
319 {
320 \tl_set:Nn \l__enumext_envir_name_tl { keyans* }
321 \tl_set:Ne \l__enumext_check_start_line_env_tl
322 {
323 in ~ 'keyans*' ~ start ~ on ~ line ~ \exp_not:V \inputlineno
324 }
325 }
326 {keyanspic}
327 {
328 \tl_set:Nn \l__enumext_envir_name_tl { keyanspic }
329 \tl_set:Ne \l__enumext_check_start_line_env_tl
330 {

```



```

331 in ~ 'keyanspic' ~ start ~ on ~ line ~ \exp_not:V \inputlineno
332 }
333 }
334 }
335 }

```

(End of definition for `\__enumext_keyans_name_and_start:`.)

### 13.5.2 Utilities for log and terminal

The function `\__enumext_reset_global_vars:` will be passed to the function `\__enumext_execute_after_env:` and will return the global variables to their default values after being used.

```

336 \cs_new_protected:Nn __enumext_reset_global_vars:
337 {
338 __enumext_reset_global_int:
339 __enumext_reset_global_bool:
340 __enumext_reset_global_tl:
341 }
342 \cs_new_protected:Nn __enumext_reset_global_int:
343 {
344 \int_gzero:N \g__enumext_item_number_int
345 \int_gzero:N \g__enumext_item_anskey_int
346 \int_gzero:N \g__enumext_item_answer_diff_int
347 }
348 \cs_new_protected:Nn __enumext_reset_global_bool:
349 {
350 \bool_gset_false:N \g__enumext_check_ans_key_bool
351 \bool_gset_false:N \g__enumext_standar_bool
352 \bool_gset_false:N \g__enumext_starred_bool
353 }
354 \cs_new_protected:Nn __enumext_reset_global_tl:
355 {
356 \tl_gclear:N \g__enumext_store_name_tl
357 \tl_gclear:N \g__enumext_start_line_tl
358 \tl_gclear:N \g__enumext_envir_name_tl
359 }

```

(End of definition for `\__enumext_reset_global_vars:` and others.)

The function `\__enumext_log_global_vars:` will be passed to the function `\__enumext_execute_after_env:` and write to the `.log` file the number of elements saved in the *prop list* and *sequence* created by the *save-ans* key along with the value of the integer variable created for the *resume* key.

```

360 \cs_new_protected:Nn __enumext_log_global_vars:
361 {
362 \msg_log:nneeee { enumext } { prop-seq-int-hook }
363 { \g__enumext_store_name_tl }
364 { \prop_count:c { g__enumext_ \g__enumext_store_name_tl _prop } }
365 { \seq_count:c { g__enumext_ \g__enumext_store_name_tl _seq } }
366 { \int_use:c { g__enumext_resume_ \g__enumext_store_name_tl _int } }
367 }

```

The function `\__enumext_log_answer_vars:` will be passed to the function `\__enumext_execute_after_env:` and write to the `.log` file the number of items and answers along with the difference between them.

```

368 \cs_new_protected:Nn __enumext_log_answer_vars:
369 {
370 \msg_log:nneee { enumext } { item-answer-hook }
371 { \int_use:N \g__enumext_item_number_int }
372 { \int_use:N \g__enumext_item_anskey_int }
373 { \int_eval:n { \g__enumext_item_number_int - \g__enumext_item_anskey_int } }
374 }

```

(End of definition for `\__enumext_log_global_vars:` and `\__enumext_log_answer_vars:`.)

## 13.6 Copying list and minipage environments

The `list` environment provided by  $\TeX$  has the following plain form:

```

\list{⟨arg one⟩}{⟨arg two⟩}
 \item[⟨opt⟩]
\endlist

```

And `minipage` environment provided by  $\TeX$  has the following (simplified) plain form:

```

\minipage[⟨pos⟩][⟨height⟩][⟨inner-pos⟩]{⟨width⟩}
 ⟨internal implement⟩
\endminipage

```

As a precaution we copy them using `\__enumext_at_begin_document:n` in case any package redefines the `\list` environment or a related command.

- For compatibility with *tagged* PDF we should use `\NewCommandCopy` and not `\cs_new_eq:NN` for `\item`. When *tagged* PDF is active `\item` is redefined using `\ltxcmd` (see `latex-lab-block[18]`).

```

__enumext_start_list:nn
__enumext_stop_list:
__enumext_item_std:w
__enumext_minipage:w
__enumext_endminipage:

```

The functions `\__enumext_start_list:nn` and `\__enumext_stop_list:` correspond to copies of `\list` and `\endlist` from plain definition of `\list` environment, the function `\__enumext_item_std:w` is a copy of the `\item` command.

```

375 __enumext_at_begin_document:n
376 {
377 \cs_new_eq:NN __enumext_start_list:nn \list
378 \cs_new_eq:NN __enumext_stop_list: \endlist
379 \NewCommandCopy __enumext_item_std:w \item
380 }

```

The functions `\__enumext_minipage:w` and `\__enumext_endminipage:` correspond to copies of `\minipage` and `\endminipage` from plain definition of `\minipage` environment.

```

381 __enumext_at_begin_document:n
382 {
383 \cs_new_eq:NN __enumext_minipage:w \minipage
384 \cs_new_eq:NN __enumext_endminipage: \endminipage
385 }

```

(End of definition for `\__enumext_start_list:nn` and others.)

### 13.7 Compatibility with hyperref and footnotehyper

```

__enumext_after_hyperref:
__enumext_hypertarget:nn
__enumext_phantomsection:

```

First we define the necessary rules using “hooks” to determine if the `hyperref` package is loaded.

```

386 \hook_gput_code:nnn { begindocument } { enumext } { __enumext_after_hyperref: }
387 \hook_gset_rule:nnnn { begindocument } { enumext } { after } { hyperref }

```

The function `\__enumext_after_hyperref:` sets the state of the boolean variable `\l__enumext_hyperref_bool` to “true” if the package is loaded. At this point we will use the public macro `\IfHyperBoolean` to determine if the `hyperfootnotes=true` key is present, if so, we set the state of the boolean variable `\__enumext_footnotes_key_bool` to “true”.

```

388 \cs_new_protected:Nn __enumext_after_hyperref:
389 {
390 \IfPackageLoadedTF { hyperref }
391 {
392 \msg_info:nnn { enumext } { package-load } { hyperref }
393 \bool_set_true:N \l__enumext_hyperref_bool
394 \IfHyperBoolean{hyperfootnotes}
395 {
396 \bool_set_true:N \l__enumext_footnotes_key_bool
397 }
398 }
399 }
400 { }

```

If the state of the variable `\l__enumext_footnotes_key_bool` is true we will check if the package `footnotehyper` is loaded, in case it is not present, we will set the value of `\l__enumext_footnotes_key_bool` to false and we will redefine `\footnote`.

```

401 \bool_if:NT \l__enumext_footnotes_key_bool
402 {
403 \IfPackageLoadedTF { footnotehyper }
404 {
405 \msg_info:nnn { enumext } { package-load } { footnotehyper }
406 }
407 {
408 \bool_set_false:N \l__enumext_footnotes_key_bool
409 }
410 }

```

The functions `\__enumext_hypertarget:nn` and `\__enumext_phantomsection:` correspond to the internal copies of `\hypertarget` and `\phantomsection`. If the boolean variable `\l__enumext_hyperref_bool` is false the functions `\__enumext_hypertarget:nn` and `\__enumext_phantomsection:` will be disabled.

```

411 \bool_if:NTF \l__enumext_hyperref_bool
412 {

```

```

413 \cs_new_eq:NN __enumext_hypertarget:nn \hypertarget
414 \cs_new_eq:NN __enumext_phantomsection: \phantomsection
415 }
416 {
417 \cs_new_eq:NN __enumext_hypertarget:nn \use_none:nn
418 \cs_new_eq:NN __enumext_phantomsection: \prg_do_nothing:
419 }
420 }

```

(End of definition for `\__enumext_after_hyperref:`, `\__enumext_hypertarget:nn`, and `\__enumext_phantomsection:`.)

`\__enumext_newlabel:nn` The function `\__enumext_newlabel:nn` write the information to the `.aux` file when using the `save-ref` key. The arguments taken by the function are:

#1: `\l__enumext_newlabel_arg_one_tl`

#2: `\l__enumext_newlabel_arg_two_tl`

• The trick here is to manage the number of arguments passed to `\newlabel{#1}{#2}` according to the presence of the `hyperref` package.

```

421 \cs_new_protected:Npn __enumext_newlabel:nn #1 #2
422 {
423 \protected@write \@auxout { }
424 {
425 \token_to_str:N \newlabel {#1}
426 {
427 {#2}
428 \bool_if:NT \l__enumext_hyperref_bool
429 { { \thepage } {#2} {#1} }
430 { }
431 }
432 }
433 __enumext_hypertarget:nn {#1} { }
434 __enumext_phantomsection:
435 }

```

(End of definition for `\__enumext_newlabel:nn`.)

### 13.8 Internal redefining `\footnote` command

To keep the correct numbering of `\footnote` and to make it work correctly in the `enumext*` and `keyans*` environments and `mini-env` key it is necessary to redefine the `\footnote` command. This implementation is adapted from the answer given by Clea F. Rees (@cfr) in [footnotes in boxes compatible with hyperref](#).

`\__enumext_footnotetext:nn` `\__enumext_renew_footnote:` `\__enumext_print_footnote:` `\__enumext_renew_footnote_mini:` `\__enumext_print_footnote_mini:` Redefinition of the `\footnote` command using `\footnotetext` and `\footnotemark` for the `mini-env` key in the `enumext` and `keyans` environments.

```

436 \cs_new_protected:Nn __enumext_footnotetext:nn
437 {
438 \footnotetext[#1]{#2}
439 }
440 \cs_new_protected:Nn __enumext_renew_footnote:
441 {
442 \RenewDocumentCommand \footnote { o +m }
443 {
444 \tl_if_novalue:nTF {##1}
445 {
446 \stepcounter{footnote}
447 \int_gset_eq:Nc \g__enumext_footnote_standar_int { c@footnote }
448 }
449 {
450 \int_gset:Nn \g__enumext_footnote_standar_int { ##1 }
451 }
452 \footnotemark [\g__enumext_footnote_standar_int]
453 \seq_gput_right:Nn \g__enumext_footnote_standar_arg_seq { ##2 }
454 \seq_gput_right:NV
455 \g__enumext_footnote_standar_int_seq \g__enumext_footnote_standar_int
456 }
457 }
458 \cs_new_protected:Nn __enumext_print_footnote:
459 {
460 \seq_if_empty:NF \g__enumext_footnote_standar_int_seq
461 {
462 \seq_map_pairwise_function:NNN

```

```

463 \g__enumext_footnote_standar_int_seq
464 \g__enumext_footnote_standar_arg_seq
465 __enumext_footnotetext:nn
466 }
467 \seq_gclear:N \g__enumext_footnote_standar_arg_seq
468 \seq_gclear:N \g__enumext_footnote_standar_int_seq
469 }

```

The `enumext*` and `keyans*` environments are implemented using `minipage` so we must also redefine `\footnote` to keep these numbering as if it were part of the document.

```

470 \cs_new_protected:Nn __enumext_renew_footnote_mini:
471 {
472 \RenewDocumentCommand \footnote { o +m }
473 {
474 \tl_if_novalue:nTF {##1}
475 {
476 \stepcounter{footnote}
477 \int_gset_eq:Nc \g__enumext_footnote_starred_int { c@footnote }
478 }
479 {
480 \int_gset:Nn \g__enumext_footnote_starred_int { ##1 }
481 }
482 \footnotemark [\g__enumext_footnote_starred_int]
483 \seq_gput_right:Nn \g__enumext_footnote_starred_arg_seq { ##2 }
484 \seq_gput_right:NV
485 \g__enumext_footnote_starred_int_seq \g__enumext_footnote_starred_int
486 }
487 }
488 \cs_new_protected:Nn __enumext_print_footnote_mini:
489 {
490 \seq_if_empty:NF \g__enumext_footnote_starred_int_seq
491 {
492 \seq_map_pairwise_function:NNN
493 \g__enumext_footnote_starred_int_seq
494 \g__enumext_footnote_starred_arg_seq
495 __enumext_footnotetext:nn
496 }
497 \seq_gclear:N \g__enumext_footnote_starred_arg_seq
498 \seq_gclear:N \g__enumext_footnote_starred_int_seq
499 }

```

(End of definition for `\__enumext_footnotetext:nn` and others.)

`\__enumext_renew_footnote_standar:` We encapsulate the redefinition of `\footnote` to pass it to internal `__enumext_mini_page` environment used by the `mini-env` key in the `enumext` and `keyans` environments. We will run the redefinition when `tagged` PDF is active or when the `footnotehyper` package is not loaded.

```

500 \cs_new_protected:Nn __enumext_renew_footnote_standar:
501 {
502 \bool_if:NT \g__enumext_standar_bool
503 {
504 \IfDocumentMetadataTF
505 {
506 __enumext_renew_footnote:
507 }
508 {
509 \bool_if:NF \l__enumext_footnotes_key_bool
510 {
511 __enumext_renew_footnote:
512 }
513 }
514 }
515 }
516 \cs_new_protected:Nn __enumext_print_footnote_standar:
517 {
518 \bool_if:NT \g__enumext_standar_bool
519 {
520 \IfDocumentMetadataTF
521 {
522 __enumext_print_footnote:
523 }
524 }

```

```

525 \bool_if:NF \l__enumext_footnotes_key_bool
526 {
527 __enumext_print_footnote:
528 }
529 }
530 }
531 }

```

We encapsulate the redefinition of `\footnote` to pass it to the `enumext*` and `keyans*` environments. We will run the redefinition when *tagged* PDF is active or when the `footnotehyper` package is not loaded.

```

532 \cs_new_protected:Nn __enumext_renew_footnote_starred:
533 {
534 \IfDocumentMetadataTF
535 {
536 __enumext_renew_footnote_mini:
537 }
538 {
539 \bool_if:NF \l__enumext_footnotes_key_bool
540 {
541 __enumext_renew_footnote_mini:
542 }
543 }
544 }
545 \cs_new_protected:Nn __enumext_print_footnote_starred:
546 {
547 \IfDocumentMetadataTF
548 {
549 __enumext_print_footnote_mini:
550 }
551 {
552 \bool_if:NF \l__enumext_footnotes_key_bool
553 {
554 __enumext_print_footnote_mini:
555 }
556 }
557 }

```

In `enumext*` and `keyans*` environments we need to use “hooks” to print `\footnote` with support for *tagged* PDF.

```

558 __enumext_after_env:nn { enumext* }
559 {
560 __enumext_print_footnote_starred:
561 }
562 __enumext_after_env:nn { keyans* }
563 {
564 __enumext_print_footnote_starred:
565 }

```

(End of definition for `\__enumext_renew_footnote_standar:` and others.)

### 13.9 The internal minipage environment

```

__enumext_internal_mini_page:
__enumext_mini_env*

```

The function `\__enumext_internal_mini_page:` creates a internal `\__enumext_mini_page` environment (*custom version of minipage*) setting the `\if@minipage` switch to “false” to allow spaces at the “above” of the environment, plus we will add `\skip_vertical:N \c_zero_skip` to maintain alignment on “top” in the first part and `\skip_vertical:N \c_zero_skip` in the second part to allow spaces “below”. This environment will be used internally by the `mini-env` key, it is NOT documented in the user interface and is for internal use only. Within this environment we redefine `\footnote` to make them look the same as if they were elsewhere in the document. This function is passed to the function `\__enumext_safe_exec:` in the `enumext` environment definition (§13.39) and `\__enumext_safe_exec_vii:` in the `enumext*` environment definition (§13.44)

```

566 \cs_new_protected:Nn __enumext_internal_mini_page:
567 {
568 \int_compare:nNtT { \l__enumext_level_int } = { 0 }
569 {
570 \DeclareDocumentEnvironment{__enumext_mini_page}{ m }
571 {
572 __enumext_renew_footnote_standar:
573 __enumext_minipage:w [t] { ##1 }
574 \legacy_if_gset_false:n { @minipage }
575 \skip_vertical:N \c_zero_skip
576 }
577 }

```

```

577 {
578 \skip_vertical:N \c_zero_skip
579 __enumext_endminipage:
580 __enumext_print_footnote_standar:
581 }
582 }
583 }

```

(End of definition for `\__enumext_internal_mini_page:` and `\__enumext_mini_env*`.)

### 13.10 Definition of public dimension

The package `enumext` only provides a single public dimension `\itemwidth` and is intended for user convenience only and is not for internal use as such. This dimension is set in all environments and is only used by the `wrap-ans` key at its default value.

```

584 \dim_zero_new:N \itemwidth

```

### 13.11 Definition of counters

```

__enumext_define_counters:Nn

```

```

enumXi

```

```

enumXii

```

```

enumXiii

```

```

enumXiv

```

```

enumXv

```

```

enumXvi

```

```

enumXvii

```

```

enumXviii

```

To create the necessary “counters” we must first make sure that they are not already defined by the user or a package such as `enumitem`, otherwise a error will be returned and the package loading will be aborted. The arguments taken by the function are:

#1: A token list `\__enumext_counter_X_tl` for “store” the counter’s name.

#2: The counter’s name.

```

585 \cs_new_protected:Npn __enumext_define_counters:Nn #1 #2
586 {
587 \cs_if_exist:cTF { c@ #2 }
588 { \msg_fatal:nnn { enumext } { counters } { #2 } }
589 {
590 \tl_set:Nn #1 { #2 }
591 \newcounter { #2 }
592 }
593 }

```

The counters created here are `enumXi`, `enumXii`, `enumXiii` and `enumXiv` for `enumext` environment, `enumXv` for `keyans` environment, `enumXvi` for `keyanspic` environment, `enumXvii` for `enumext*` and `enumXviii` for the `keyans*` environments.

```

594 __enumext_define_counters:Nn __enumext_counter_i_tl { enumXi }
595 __enumext_define_counters:Nn __enumext_counter_ii_tl { enumXii }
596 __enumext_define_counters:Nn __enumext_counter_iii_tl { enumXiii }
597 __enumext_define_counters:Nn __enumext_counter_iv_tl { enumXiv }
598 __enumext_define_counters:Nn __enumext_counter_v_tl { enumXv }
599 __enumext_define_counters:Nn __enumext_counter_vi_tl { enumXvi }
600 __enumext_define_counters:Nn __enumext_counter_vii_tl { enumXvii }
601 __enumext_define_counters:Nn __enumext_counter_viii_tl { enumXviii }

```

(End of definition for `\__enumext_define_counters:Nn` and others.)

### 13.12 Definition of labels

This part of the code is inspired by the `enumitem` package. The idea is to be able to access the counters using `\arabic*`, `\Alph*`, `\alph*`, `\Roman*` and `\roman*` to use them in the `label` key.

```

__enumext_register_counter_style:Nn

```

These `<counters>` will be used as default `<labels>` if the `label` key is not used for the different levels of the `enumext`, `enumext*`, `keyans` and `keyans*` environments, so it is necessary to get a default value for `labelwidth` from these `<labels>` at the same time.

```

602 \cs_new_protected:Npn __enumext_register_counter_style:Nn #1 #2
603 {
604 \tl_const:cn { c__enumext_widest_ \cs_to_str:N #1 _tl } {#2}
605 \tl_gput_right:Nn \g__enumext_counter_styles_tl {#1}
606 }
607 __enumext_register_counter_style:Nn \arabic { 0 }
608 __enumext_register_counter_style:Nn \Alph { M }
609 __enumext_register_counter_style:Nn \alph { m }
610 __enumext_register_counter_style:Nn \Roman { VIII }
611 __enumext_register_counter_style:Nn \roman { viii }

```

(End of definition for `\__enumext_register_counter_style:Nn`.)



`\__enumext_label_width_by_box:Nn`  
`\__enumext_label_width_by_box:cv`

The function `\__enumext_label_width_by_box:Nn` set the default `\labelwidth` using a box width if no `\labelwidth` key is passed.

```

612 \cs_new_protected:Npn __enumext_label_width_by_box:Nn #1 #2
613 {
614 \hbox_set:Nn \l__enumext_label_width_by_box {#2}
615 \dim_set:Nn #1 { \box_wd:N \l__enumext_label_width_by_box }
616 }
617 \cs_generate_variant:Nn __enumext_label_width_by_box:Nn { cv }

```

(End of definition for `\__enumext_label_width_by_box:Nn`.)

`\__enumext_label_style:Nnn`  
`\__enumext_label_style:cvn`

The function `\__enumext_label_style:Nnn` is used by the `\label` key to creates the variables containing the `\label style` and will allow to use `\arabic*`, `\Alph*`, `\alph*`, `\Roman*` and `\roman*` as arguments. It loops through the defined counter styles in `\g__enumext_counter_styles_tl` (`\arabic`, `\alph`, `\Alph`, `\roman`, and `\Roman`) for example, looking for `\roman*` and replacing that by `\roman{<counter>}`, and doing the same for the `\g__enumext_widest_label_tl` to keep both in sync.

```

618 \cs_new_protected:Npn __enumext_label_style:Nnn #1 #2 #3
619 {
620 \tl_clear_new:N #1
621 \tl_put_right:Ne #1 { \tl_trim_spaces:n {#3} }
622 \tl_gset_eq:NN \g__enumext_widest_label_tl #1
623 \tl_map_inline:Nn \g__enumext_counter_styles_tl
624 {
625 \tl_replace_all:Nne #1 { ##1* } { \exp_not:N ##1 {#2} }
626 \tl_greplace_all:Nne \g__enumext_widest_label_tl { ##1* }
627 { \tl_use:c { c__enumext_widest_ \cs_to_str:N ##1 _tl } }
628 }
629 __enumext_label_width_by_box:Nn \l__enumext_current_widest_dim
630 { \tl_use:N \g__enumext_widest_label_tl }
631 \tl_set_eq:cN { the #2 } #1
632 }
633 \cs_generate_variant:Nn __enumext_label_style:Nnn { cvn }

```

(End of definition for `\__enumext_label_style:Nnn`.)

### 13.13 Setting keys associated with label

When `tagged PDF` is active `\makelabel` is redefined using `\makebox` to work correctly (§13.34). From the user side it is convenient to have a key that allows using this redefinition with `\makebox` without having `\IfDocumentMetadataTF` active.

`mode-box`

We define the key `mode-box` only for the “first level” of `enumext` and `enumext*` environments.

```

634 \cs_set_protected:Npn __enumext_tmp:n #1
635 {
636 \keys_define:nn { enumext / #1 }
637 {
638 mode-box .bool_set:N = \l__enumext_mode_box_bool,
639 mode-box .initial:n = false,
640 mode-box .value_forbidden:n = true,
641 }
642 }
643 \clist_map_inline:nn { level-1, enumext* } { __enumext_tmp:n {#1} }

```

(End of definition for `mode-box`.)

`font`  
`labelsep`  
`labelwidth`  
`wrap-label`  
`wrap-label*`

Definition of keys `font`, `labelsep`, `labelwidth`, `wrap-label` and `wrap-label*` keys for `enumext` and `keyans` environments.

```

644 \cs_set_protected:Npn __enumext_tmp:nn #1 #2
645 {
646 \keys_define:nn { enumext / #1 }
647 {
648 font .tl_set:c = { \l__enumext_label_font_style_#2_tl },
649 font .value_required:n = true,
650 labelsep .dim_set:c = { \l__enumext_labelsep_#2_dim },
651 labelsep .initial:n = {0.3333em},
652 labelsep .value_required:n = true,
653 labelwidth .dim_set:c = { \l__enumext_labelwidth_#2_dim },
654 labelwidth .value_required:n = true,
655 wrap-label .cs_set_protected:cp = { __enumext_wrapper_label_#2:n } ##1,
656 wrap-label .initial:n = {##1},
657 wrap-label .value_required:n = true,

```

```

658 wrap-label* .code:n = {
659 \bool_set_true:c { l__enumext_wrap_label_opt_#2_bool }
660 \keys_set:nn { enumext / #1 } { wrap-label = {##1} }
661 },
662 wrap-label* .value_required:n = true,
663 }
664 }
665 \clist_map_inline:Nn \c__enumext_all_envs_clist { __enumext_tmp:nn #1 }

```

(End of definition for font and others.)

`align` The `align` key is implemented differently for “starred” and “non starred” environments. For compatibility with tagged PDF we must set `\l__enumext_align_label_pos_X_str`.

```

666 \cs_set_protected:Npn __enumext_tmp:nn #1 #2
667 {
668 \keys_define:nn { enumext / #1 }
669 {
670 align .choice:,
671 align / left .code:n =
672 {
673 \tl_clear:c { l__enumext_label_fill_left_#2_tl }
674 \tl_set:cn { l__enumext_label_fill_right_#2_tl } { \hfill }
675 \str_set:cn { l__enumext_align_label_pos_#2_str } { l }
676 },
677 align / right .code:n =
678 {
679 \tl_set:cn { l__enumext_label_fill_left_#2_tl } { \hfill }
680 \tl_clear:c { l__enumext_label_fill_right_#2_tl }
681 \str_set:cn { l__enumext_align_label_pos_#2_str } { r }
682 },
683 align / center .code:n =
684 {
685 \tl_set:cn { l__enumext_label_fill_left_#2_tl } { \hfill }
686 \tl_set:cn { l__enumext_label_fill_right_#2_tl } { \hfill }
687 \str_set:cn { l__enumext_align_label_pos_#2_str } { c }
688 },
689 align / unknown .code:n =
690 \msg_error:nneee { enumext } { unknown-choice }
691 { align } { left, ~ right, ~ center } { \exp_not:n {##1} },
692 align .initial:n = left,
693 align .value_required:n = true,
694 }
695 }
696 \clist_map_inline:nn
697 {
698 {level-1}{i}, {level-2}{ii}, {level-3}{iii}, {level-4}{iv}, {keyans}{v}
699 }
700 { __enumext_tmp:nn #1 }
701
702 \cs_set_protected:Npn __enumext_tmp:nn #1 #2
703 {
704 \keys_define:nn { enumext / #1 }
705 {
706 align .choice:,
707 align / left .code:n = \str_set:cn { l__enumext_align_label_#2_str } { l },
708 align / right .code:n = \str_set:cn { l__enumext_align_label_#2_str } { r },
709 align / center .code:n = \str_set:cn { l__enumext_align_label_#2_str } { c },
710 align / unknown .code:n =
711 \msg_error:nneee { enumext } { unknown-choice }
712 { align } { left, ~ right, ~ center } { \exp_not:n {##1} },
713 align .initial:n = left,
714 align .value_required:n = true,
715 }
716 }
717 \clist_map_inline:nn { {enumext*}{vii}, {keyans*}{viii} } { __enumext_tmp:nn #1 }

```

(End of definition for align.)

### 13.14 Setting label and ref keys

The implementation of the keys `label` and `ref` are part of the core of the package `enumext`, here the default values for `\label`, the value of the variables `\l__enumext_label_X_tl`, the default values for `\labelwidth` and the “label and ref” system.

### 13.14.1 Define and set label and ref keys for enumext environment

label Here we set the default *⟨labels⟩* of the *four levels* of `enumext` environment, along with the default value for `labelwidth` key and `ref` key.

```

\l__enumext_label_i_tl 717 \cs_set_protected:Npn __enumext_tmp:nnn #1 #2 #3
\l__enumext_label_ii_tl 718 {
\l__enumext_label_iii_tl 719 \keys_define:nn { enumext / #1 }
\l__enumext_label_iv_tl 720 {
721 label .code:n = {
722 __enumext_label_style:cvn { l__enumext_label_#2_tl }
723 { l__enumext_counter_#2_tl } {##1}
724 \dim_set_eq:cN { l__enumext_labelwidth_#2_dim }
725 \l__enumext_current_widest_dim
726 },
727 label .initial:n = #3,
728 label .value_required:n = true,
729 ref .code:n = __enumext_standar_ref:n {##1},
730 ref .value_required:n = true,
731 }
732 }
733 __enumext_tmp:nnn { level-1 } { i } { \arabic*. }
734 __enumext_tmp:nnn { level-2 } { ii } { (\alph*) }
735 __enumext_tmp:nnn { level-3 } { iii } { \roman*. }
736 __enumext_tmp:nnn { level-4 } { iv } { \Alph*. }

```

(End of definition for `label` and others.)

\\_\_enumext\_standar\_ref:n The \\_\_enumext\_standar\_ref:n first we will pass the key argument to \\_\_enumext\_ref\_key\_arg\_tl and we will analyze its state, if it is not *empty* we will make a copy of the current counter in \\_\_enumext\_ref\_the\_count\_tl and we will execute the function \\_\_enumext\_regex\_counter\_style: which will return the modified \\_\_enumext\_ref\_key\_arg\_tl and we make the value of \\_\_enumext\_ref\_the\_count\_tl the same as that \\_\_enumext\_the\_counter\_X\_tl which contains \theenumX and finally we set \\_\_enumext\_renew\_the\_count\_X\_tl with the renewed command.

```

737 \cs_new_protected:Npn __enumext_standar_ref:n #1
738 {
739 \tl_set:Nn __enumext_ref_key_arg_tl {#1}
740 \tl_if_empty:NTF __enumext_ref_key_arg_tl
741 {
742 \msg_error:nnn { enumext } { key-ref-empty } { enumext }
743 }
744 {
745 \tl_set_eq:Nc
746 __enumext_ref_the_count_tl { l__enumext_counter_ __enumext_level: _tl }
747 __enumext_regex_counter_style:
748 \tl_set_eq:Nc
749 __enumext_ref_the_count_tl { l__enumext_the_counter_ __enumext_level: _tl }
750 \tl_put_right:ce { l__enumext_renew_the_count_ __enumext_level: _tl }
751 {
752 \exp_not:N \renewcommand { \exp_not:V __enumext_ref_the_count_tl } { \exp_not:V \l__
753 }
754 }
755 }

```

Finally the function \\_\_enumext\_standar\_ref: will execute the modification for the reference system in the second argument of the environment definition `enumext`.

```

756 \cs_new_protected:Nn __enumext_standar_ref:
757 {
758 \tl_if_empty:cF { l__enumext_renew_the_count_ __enumext_level: _tl }
759 {
760 \tl_use:c { l__enumext_renew_the_count_ __enumext_level: _tl }
761 }
762 }

```

(End of definition for \\_\_enumext\_standar\_ref:n and \\_\_enumext\_standar\_ref:.)

### 13.14.2 Define and set label and ref keys for enumext\* and keyans\* environments

label Here we set the default *⟨labels⟩* for `enumext*` and `keyans*` environments, along with the default value for `labelwidth` key and `ref` key.

```

\l__enumext_label_vii_tl 763 \cs_set_protected:Npn __enumext_tmp:nnn #1 #2 #3
\l__enumext_label_viii_tl 764 {
765 \keys_define:nn { enumext / #1 }

```

```

766 {
767 label .code:n = {
768 __enumext_label_style:cvn { l__enumext_label_#2_tl }
769 { l__enumext_counter_#2_tl } {##1}
770 \dim_set_eq:cN { l__enumext_labelwidth_#2_dim }
771 \l__enumext_current_widest_dim
772 },
773 label .initial:n = #3,
774 label .value_required:n = true,
775 ref .code:n = __enumext_starred_ref:n {##1},
776 ref .value_required:n = true,
777 }
778 }
779 __enumext_tmp:nnn { enumext* } { vii } { \arabic*.}
780 __enumext_tmp:nnn { keyans* } { viii } { \Alph*.}

```

(End of definition for `label` and others.)

`\__enumext_starred_ref:n` The implementation of `\__enumext_starred_ref:n` is the same as that used for the environment `enumext`.

```

__enumext_starred_ref:
781 \cs_new_protected:Npn __enumext_starred_ref:n #1
782 {
783 \tl_set:Nn \l__enumext_ref_key_arg_tl {#1}
784 \int_compare:nNnT { \l__enumext_level_h_int } = { 1 }
785 {
786 \tl_if_empty:NTF \l__enumext_ref_key_arg_tl
787 {
788 \msg_error:nnn { enumext } { key-ref-empty } { enumext* }
789 }
790 {
791 \tl_set_eq:NN \l__enumext_ref_the_count_tl \l__enumext_counter_vii_tl
792 __enumext_regex_counter_style:
793 \tl_set_eq:NN \l__enumext_ref_the_count_tl \l__enumext_the_counter_vii_tl
794 \tl_put_right:Ne \l__enumext_renew_the_count_vii_tl
795 {
796 \exp_not:N \renewcommand { \exp_not:V \l__enumext_ref_the_count_tl } { \exp_not:V
797 }
798 }
799 }
800 \int_compare:nNnT { \l__enumext_keyans_level_h_int } = { 1 }
801 {
802 \tl_if_empty:NTF \l__enumext_ref_key_arg_tl
803 {
804 \msg_error:nnn { enumext } { key-ref-empty } { keyans* }
805 }
806 {
807 \tl_set_eq:NN \l__enumext_ref_the_count_tl \l__enumext_counter_viii_tl
808 __enumext_regex_counter_style:
809 \tl_set_eq:NN \l__enumext_ref_the_count_tl \l__enumext_the_counter_viii_tl
810 \tl_put_right:Ne \l__enumext_renew_the_count_viii_tl
811 {
812 \exp_not:N \renewcommand { \exp_not:V \l__enumext_ref_the_count_tl } { \exp_not:V
813 }
814 }
815 }
816 }

```

Finally the function `\__enumext_starred_ref:` will execute the modification for the reference system in the second argument of the `enumext*` and `keyans*` environment definition.

```

817 \cs_new_protected:Npn __enumext_starred_ref:
818 {
819 \int_compare:nNnT { \l__enumext_level_h_int } = { 1 }
820 {
821 \tl_if_empty:NF \l__enumext_renew_the_count_vii_tl
822 {
823 \tl_use:N \l__enumext_renew_the_count_vii_tl
824 }
825 }
826 \int_compare:nNnT { \l__enumext_keyans_level_h_int } = { 1 }
827 {
828 \tl_if_empty:NF \l__enumext_renew_the_count_viii_tl
829 {
830 \tl_use:N \l__enumext_renew_the_count_viii_tl

```

```

831 }
832 }
833 }

```

(End of definition for `\__enumext_starred_ref:n` and `\__enumext_starred_ref:.`)

### 13.14.3 Define and set label and ref keys for keyans and keyanspic environments

Here we set the default *label* for `keyans` and `keyanspic` environment, along with the default value for `labelwidth` and `ref` key. The `keyanspic` environment use the same *label* as the `keyans` environment.

```

__enumext_label_v_tl 834 \keys_define:nn { enumext / keyans }
__enumext_label_vi_tl 835 {
836 label .code:n = {
837 __enumext_label_style:cvn { __enumext_label_v_tl }
838 { __enumext_counter_v_tl } {#1}
839 \dim_set_eq:cN { __enumext_labelwidth_v_dim }
840 __enumext_current_widest_dim
841 __enumext_label_style:cvn { __enumext_label_vi_tl }
842 { __enumext_counter_vi_tl } {#1}
843 \dim_set_eq:cN { __enumext_labelwidth_v_dim }
844 __enumext_current_widest_dim
845 },
846 label .initial:n = \Alph*,
847 label .value_required:n = true,
848 ref .code:n = __enumext_keyans_ref:n {#1},
849 ref .value_required:n = true,
850 }

```

(End of definition for `label` and others.)

The implementation of `\__enumext_keyans_ref:n` is the same as that used for the environment `enumext`.

```

__enumext_keyans_ref:n 851 \cs_new_protected:Npn __enumext_keyans_ref:n #1
__enumext_keyans_ref: 852 {
853 \tl_set:Nn __enumext_ref_key_arg_tl {#1}
854 \tl_if_empty:NTF __enumext_ref_key_arg_tl
855 {
856 \msg_error:nnn { enumext } { key-ref-empty } { keyans }
857 }
858 {
859 \tl_set_eq:NN __enumext_ref_the_count_tl __enumext_counter_v_tl
860 __enumext_regex_counter_style:
861 \tl_set_eq:NN __enumext_ref_the_count_tl __enumext_the_counter_v_tl
862 \tl_put_right:Ne __enumext_renew_the_count_v_tl
863 {
864 \exp_not:N \renewcommand { \exp_not:V __enumext_ref_the_count_tl } { \exp_not:V \l_
865 }
866 }
867 }

```

Finally the function `\__enumext_keyans_ref:` will execute the modification for the reference system in the second argument of the `keyans*` environment definition.

```

868 \cs_new_protected:Nn __enumext_keyans_ref:
869 {
870 \tl_if_empty:NF __enumext_renew_the_count_v_tl
871 {
872 \tl_use:N __enumext_renew_the_count_v_tl
873 }
874 }

```

(End of definition for `\__enumext_keyans_ref:n` and `\__enumext_keyans_ref:.`)

### 13.15 Setting start, start\* and widest keys

The function `\__enumext_start_from:NNn` used by `start` and `start*` keys take three arguments:

```

__enumext_start_from:ccn #1: __enumext_label_X_tl
__enumext_start_from:cce #2: __enumext_start_X_int
__enumext_start_from:cce #3: <integer or string>

```

The first argument of this function are the “*counter style*” set by `label` key, the second argument is returned by the function, the third argument can be an *integer* or *string* of the form `\Alph`, `\alph`, `\Roman` or `\roman`. This effectively allows `start=A` or `start=1` to be used.

```

875 \cs_new_protected:Npn __enumext_start_from:NNn #1 #2 #3
876 {

```

```

877 __enumext_if_is_int:nTF { #3 }
878 {
879 \int_set:Nn #2 {#3}
880 }
881 {
882 \regex_match:nVT { \c{Alpha} | \c{alpha} } {#1}
883 { \int_set:Nn #2 { \int_from_alph:n {#3} } }
884 \regex_match:nVT { \c{Roman} | \c{roman} } {#1}
885 { \int_set:Nn #2 { \int_from_roman:n {#3} } }
886 }
887 }
888 \cs_generate_variant:Nn __enumext_start_from:NNn { ccn, cce }

```

(End of definition for \\_\_enumext\_start\_from:NNn.)

```

__enumext_widest_from:nNNn
__enumext_widest_from:nccn

```

The function \\_\_enumext\_widest\_from:nNNn used by the `widest` key take four arguments:

- #1: The counter associated with the environment level
- #2: \\_\_enumext\_label\_X\_tl
- #3: \\_\_enumext\_labelwidth\_X\_dim
- #4: *integer or string*

The second and third arguments of this function are the values set by `label` and `labelwidth` keys, the four argument can be an *integer* or *string* of the form `\Alpha`, `\alpha`, `\Roman` or `\roman`. The value of the four argument is set temporarily for the identified counter in this point (level), then the value is expanded into a “box” and the “width” of the “box” is returned.

```

889 \cs_new_protected:Npn __enumext_widest_from:nNNn #1 #2 #3 #4
890 {
891 __enumext_if_is_int:nTF {#4}
892 {
893 \setcounter{enumX#1} { #4 }
894 }
895 {
896 \regex_match:nVT { \c{Alpha} | \c{alpha} } {#2}
897 { \setcounter{enumX#1} { \int_from_alph:n {#4} } }
898 \regex_match:nVT { \c{Roman} | \c{roman} } {#2}
899 { \setcounter{enumX#1} { \int_from_roman:n {#4} } }
900 }
901 __enumext_label_width_by_box:cv
902 { __enumext_labelwidth_#1_dim } { __enumext_label_#1_tl }
903 }
904 \cs_generate_variant:Nn __enumext_widest_from:nNNn { nccn }

```

(End of definition for \\_\_enumext\_widest\_from:nNNn.)

Now define and set `start*`, `start` and `widest` keys for `enumext`, `enumext*`, `keyans` and `keyans*` environments.

`start*`

`start`

`widest`

```

905 \cs_set_protected:Npn __enumext_tmp:nn #1 #2
906 {
907 \keys_define:nn { enumext / #1 }
908 {
909 start* .code:n = {
910 __enumext_start_from:ccn
911 { __enumext_label_#2_tl }
912 { __enumext_start_#2_int } {##1}
913 },
914 start* .value_required:n = true,
915 start .code:n = {
916 __enumext_start_from:cce
917 { __enumext_label_#2_tl }
918 { __enumext_start_#2_int } { \int_eval:n {##1} }
919 },
920 start .initial:n = 1,
921 start .value_required:n = true,
922 widest .code:n = {
923 __enumext_widest_from:nccn {#2}
924 { __enumext_label_#2_tl }
925 { __enumext_labelwidth_#2_dim } {##1}
926 },
927 widest .value_required:n = true,
928 }
929 }
930 \clist_map_inline:Nn \c__enumext_all_envs_clist { __enumext_tmp:nn #1 }

```



(End of definition for `start`, `start*`, and `widest`.)

### 13.16 Setting keys for vertical spaces

Define and set `topsep`, `partopsep`, `parsep`, `itemsep`, `noitemsep` and `nosep` keys for `enumext`, `enumext*`, `keyans` and `keyans*` environments.

```

topsep
partopsep
parsep
noitemsep
nosep
931 \cs_set_protected:Npn __enumext_tmp:nnnnnn #1 #2 #3 #4 #5 #6
932 {
933 \keys_define:nn { enumext / #1 }
934 {
935 topsep .skip_set:c = { l__enumext_topsep_#2_skip },
936 topsep .initial:n = {#3},
937 topsep .value_required:n = true,
938 partopsep .skip_set:c = { l__enumext_partopsep_#2_skip },
939 partopsep .initial:n = {#4},
940 partopsep .value_required:n = true,
941 parsep .skip_set:c = { l__enumext_parsep_#2_skip },
942 parsep .initial:n = {#5},
943 parsep .value_required:n = true,
944 itemsep .skip_set:c = { l__enumext_itemsep_#2_skip },
945 itemsep .initial:n = {#6},
946 itemsep .value_required:n = true,
947 noitemsep .meta:n = { itemsep = 0pt, parsep = 0pt },
948 noitemsep .value_forbidden:n = true,
949 nosep .meta:n = {
950 itemsep = 0pt, parsep = 0pt,
951 topsep = 0pt, partopsep = 0pt,
952 },
953 nosep .value_forbidden:n = true,
954 }
955 }
```

Now we set the values based on standard `article` class in `10pt`.

```

956 __enumext_tmp:nnnnnn { level-1 } { i } { 8.0pt plus 2.0pt minus 4.0pt }
957 { 2.0pt plus 1.0pt minus 1.0pt } { 4.0pt plus 2.0pt minus 1.0pt }
958 { 4.0pt plus 2.0pt minus 1.0pt }
959 __enumext_tmp:nnnnnn { level-2 } { ii } { 4.0pt plus 2.0pt minus 1.0pt }
960 { 2.0pt plus 1.0pt minus 1.0pt } { 2.0pt plus 1.0pt minus 1.0pt }
961 { 2.0pt plus 1.0pt minus 1.0pt }
962 __enumext_tmp:nnnnnn { level-3 } { iii } { 2.0pt plus 1.0pt minus 1.0pt }
963 { 1.0pt minus 1.0pt } { 0pt } { 2.0pt plus 1.0pt minus 1.0pt }
964 __enumext_tmp:nnnnnn { level-4 } { iv } { 2.0pt plus 1.0pt minus 1.0pt }
965 { 1.0pt minus 1.0pt } { 0pt } { 2.0pt plus 1.0pt minus 1.0pt }
966 __enumext_tmp:nnnnnn { keyans } { v } { 4.0pt plus 2.0pt minus 1.0pt }
967 { 2.0pt plus 1.0pt minus 1.0pt } { 2.0pt plus 1.0pt minus 1.0pt }
968 { 2.0pt plus 1.0pt minus 1.0pt }
969 __enumext_tmp:nnnnnn { enumext* } { vii } { 8.0pt plus 2.0pt minus 4.0pt }
970 { 2.0pt plus 1.0pt minus 1.0pt } { 4.0pt plus 2.0pt minus 1.0pt }
971 { 4.0pt plus 2.0pt minus 1.0pt }
972 __enumext_tmp:nnnnnn { keyans* } { viii } { 4.0pt plus 2.0pt minus 1.0pt }
973 { 2.0pt plus 1.0pt minus 1.0pt } { 2.0pt plus 1.0pt minus 1.0pt }
974 { 2.0pt plus 1.0pt minus 1.0pt }
```

(End of definition for `topsep` and others.)

### 13.17 Setting base-fix key

When nesting starting right after `\item` (without material between them) there is a problem with the alignment of the *baseline* between the two environments. One way to get around this problem is to place `\mode_leave_vertical:` apply `\vspace[-\baselineskip]` and set `\topsep=0pt` for the “first level” of the nested `enumext` environment.

```

base-fix
__enumext_nested_base_line_fix:
975 \keys_define:nn { enumext / level-1 }
976 {
977 base-fix .bool_set:N = \l__enumext_base_line_fix_bool,
978 base-fix .initial:n = false,
979 base-fix .value_forbidden:n = true,
980 }
```

The function `\__enumext_nested_base_line_fix:` passed to the `\__enumext_parse_keys:n` function in the definition of the `enumext` environment (§13.39) will be responsible for applying the *baseline correction* and adjusting the *⟨keys⟩* for the `enumext` environment and the `\printkeyans` with *starred argument* ‘\*’ (§13.47).

We will first implement the function code from the user side of the `base-fix` key, that is, only the user knows when it is necessary to apply it within the document in which case the variable `\__enumext_print_keyans_star_bool` set by the `\printkeyans` command is false and the variable `\__enumext_base_line_fix_bool` is true.

```

981 \cs_new_protected:Nn __enumext_nested_base_line_fix:
982 {
983 \bool_lazy_all:nT
984 {
985 { \bool_if_p:N __enumext_starred_first_bool }
986 { \bool_if_p:N __enumext_base_line_fix_bool }
987 { \bool_not_p:n { __enumext_print_keyans_star_bool } }
988 }
989 {
990 \mode_leave_vertical:
991 \vspace { -\dim_eval:n { \baselineskip + \parsep } }
992 }

```

When we are running the `\printkeyans` command with the *starred argument* ‘\*’ the variable `\__enumext_print_keyans_star_bool` is true and we can run a simplified version of `\vspace` using `\skip_vertical:n`.

```

993 \bool_lazy_and:nnT
994 { \bool_if_p:N __enumext_starred_first_bool }
995 { \bool_if_p:N __enumext_print_keyans_star_bool }
996 {
997 \mode_leave_vertical:
998 \skip_vertical:n { -\baselineskip }
999 \skip_vertical:N \c_zero_skip
1000 }

```

Finally we set the values of the keys `topsep`, `above` and `above*` for the “first level” of `enumext` environment equal to `0pt` and set the variable `\__enumext_base_line_fix_bool` to false.

```

1001 \keys_set:nn { enumext / level-1 }
1002 {
1003 topsep = 0pt, above = 0pt, above* = 0pt,
1004 }
1005 \bool_set_false:N __enumext_base_line_fix_bool
1006 }

```

(End of definition for `base-fix` and `\__enumext_nested_base_line_fix:`.)

### 13.18 Setting keys for horizontal spaces

Define and set `itemindent`, `rightmargin`, `listparindent`, `list-offset` and `list-indent` keys for `enumext`, `enumext*`, `keyans` and `keyans*` environments.

```

1007 \cs_set_protected:Npn __enumext_tmp:nn #1 #2
1008 {
1009 \keys_define:nn { enumext / #1 }
1010 {
1011 itemindent .dim_set:c = { __enumext_fake_item_indent_#2_dim },
1012 itemindent .value_required:n = true,
1013 rightmargin .dim_set:c = { __enumext_rightmargin_#2_dim },
1014 rightmargin .value_required:n = true,
1015 listparindent .dim_set:c = { __enumext_listparindent_#2_dim },
1016 listparindent .value_required:n = true,
1017 list-offset .dim_set:c = { __enumext_listoffset_#2_dim },
1018 list-offset .value_required:n = true,
1019 list-indent .code:n =
1020 \bool_set_true:c { __enumext_leftmargin_tmp_#2_bool }
1021 \dim_set:cn { __enumext_leftmargin_tmp_#2_dim } {##1},
1022 list-indent .value_required:n = true,
1023 }
1024 }
1025 \clist_map_inline:nn
1026 {
1027 {level-1}{i}, {level-2}{ii}, {level-3}{iii}, {level-4}{iv}, {keyans}{v}
1028 }
1029 { __enumext_tmp:nn #1 }

```

(End of definition for `itemindent` and others.)

For `enumext*` and `keyans*` environments the situation is a bit different, the `list-indent` key behaves like the `list-offset` key.

```

1030 \cs_set_protected:Npn __enumext_tmp:nn #1 #2
1031 {
1032 \keys_define:nn { enumext / #1 }
1033 {
1034 itemindent .dim_set:c = { l__enumext_fake_item_indent_#2_dim },
1035 itemindent .value_required:n = true,
1036 rightmargin .dim_set:c = { l__enumext_rightmargin_#2_dim },
1037 rightmargin .value_required:n = true,
1038 listparindent .dim_set:c = { l__enumext_listparindent_#2_dim },
1039 listparindent .value_required:n = true,
1040 list-offset .dim_set:c = { l__enumext_listoffset_#2_dim },
1041 list-offset .value_required:n = true,
1042 list-indent .meta:n = { list-offset = ##1 },
1043 list-indent .value_required:n = true,
1044 }
1045 }
1046 \clist_map_inline:nn
1047 {
1048 {enumext*}{vii}, {keyans*}{viii}
1049 }
1050 { __enumext_tmp:nn #1 }

```

### 13.18.1 Functions for setting the fake `itemindent`

The `itemindent` key does not set the value of `\itemindent`, it only sets the value of the *horizontal space* applied using `\skip_horizontal:N`. We will store this value in the variable and only apply it when it is greater than `0pt`. Here I will need to place `\mode_leave_vertical:` and the plain TeX macro `\ignorespaces` to avoid unwanted extra space when using the `itemindent` key.

```

1051 \cs_set_protected:Nn __enumext_fake_item_indent:
1052 {
1053 \dim_compare:nNnT
1054 { \dim_use:c { l__enumext_fake_item_indent_ __enumext_level: _dim } }
1055 >
1056 { \c_zero_dim }
1057 {
1058 \tl_set:ce { l__enumext_fake_item_indent_ __enumext_level: _tl }
1059 {
1060 \exp_not:N \mode_leave_vertical:
1061 \exp_not:n { \skip_horizontal:n }
1062 { \dim_use:c { l__enumext_fake_item_indent_ __enumext_level: _dim } }
1063 \exp_not:N \ignorespaces
1064 }
1065 }
1066 }
1067 \cs_set_protected:Nn __enumext_keyans_fake_item_indent:
1068 {
1069 \dim_compare:nNnT
1070 { \l__enumext_fake_item_indent_v_dim } > { \c_zero_dim }
1071 {
1072 \tl_set:Nc \l__enumext_fake_item_indent_v_tl
1073 {
1074 \exp_not:N \mode_leave_vertical:
1075 \exp_not:N \skip_horizontal:N \l__enumext_fake_item_indent_v_dim
1076 \exp_not:N \ignorespaces
1077 }
1078 }
1079 }
1080 \cs_set_protected:Nn __enumext_fake_item_indent_vii:
1081 {
1082 \dim_compare:nNnT
1083 { \l__enumext_fake_item_indent_vii_dim } > { \c_zero_dim }
1084 {
1085 \tl_set:Nc \l__enumext_fake_item_indent_vii_tl
1086 {
1087 \exp_not:N \skip_horizontal:N \l__enumext_fake_item_indent_vii_dim
1088 \exp_not:N \ignorespaces
1089 }
1090 }

```

```

1091 }
1092 \cs_set_protected:Nn __enumext_fake_item_indent_viii:
1093 {
1094 \dim_compare:nNt
1095 { \l__enumext_fake_item_indent_viii_dim } > { \c_zero_dim }
1096 {
1097 \tl_set:Nc \l__enumext_fake_item_indent_viii_tl
1098 {
1099 \exp_not:N \skip_horizontal:N \l__enumext_fake_item_indent_viii_dim
1100 \exp_not:N \ignorespaces
1101 }
1102 }
1103 }

```

(End of definition for `\__enumext_fake_item_indent:` and others.)

### 13.19 Setting show-length key

`show-length` Define and set `show-length` key for `enumext`, `enumext*`, `keyans` and `keyans*` environments. The function sets the boolean variable `\l__enumext_show_length_X_bool` used in the definition of all environments to “true” and calls the function `\__enumext_show_length:nnn` which prints all the values of the “vertical” and “horizontal” parameters calculated and used.

```

1104 \cs_set_protected:Npn __enumext_tmp:nn #1 #2
1105 {
1106 \keys_define:nn { enumext / #1 }
1107 {
1108 show-length .bool_set:c = { \l__enumext_show_length_#2_bool },
1109 show-length .initial:n = false,
1110 }
1111 }
1112 \clist_map_inline:Nn \c__enumext_all_envs_clist { __enumext_tmp:nn #1 }

```

(End of definition for `show-length`.)

### 13.20 Setting before, after and first keys

`before` Define and set `before`, `before*`, `after` and `first` keys for `enumext`, `enumext*`, `keyans` and `keyans*`  
`before*` environments.

```

1113 \cs_set_protected:Npn __enumext_tmp:nn #1 #2
1114 {
1115 \keys_define:nn { enumext / #1 }
1116 {
1117 before .tl_set:c = { \l__enumext_before_no_starred_key_#2_tl },
1118 before .value_required:n = true,
1119 before* .tl_set:c = { \l__enumext_before_starred_key_#2_tl },
1120 before* .value_required:n = true,
1121 after .tl_set:c = { \l__enumext_after_stop_list_#2_tl },
1122 after .value_required:n = true,
1123 first .tl_set:c = { \l__enumext_after_list_args_#2_tl },
1124 first .value_required:n = true,
1125 }
1126 }
1127 \clist_map_inline:Nn \c__enumext_all_envs_clist { __enumext_tmp:nn #1 }

```

(End of definition for `before` and others.)

#### 13.20.1 Functions for before, after and first keys in enumext

`\__enumext_before_args_exec:` The function `\__enumext_before_args_exec:` executes the `{\code}` set by the `before*` key “before” the `enumext` environment is started. The `{\code}` is executed “without” knowing any definition of the `{\arg two}` of the list: `{\code}\list{\arg one}{\arg two}`.

```

1128 \cs_new_protected:Nn __enumext_before_args_exec:
1129 {
1130 \tl_use:c { \l__enumext_before_starred_key_ __enumext_level: _tl }
1131 }

```

The function `\__enumext_before_keys_exec:` executes the `{\code}` set by the `before` key “before” the `enumext` environment is started in *second argument* of the list. The `{\code}` is executed “knowing” all definition and values provides by `\keys: \list{\arg one}{\arg two}{\code}`

```

1132 \cs_new_protected:Nn __enumext_before_keys_exec:
1133 {
1134 \tl_use:c { \l__enumext_before_no_starred_key_ __enumext_level: _tl }
1135 }

```

The function `\__enumext_after_stop_list:` executes the `{\code}` set by the `after` key “after” the `enumext` environment has finished: `\endlist{\code}`.

```
1136 \cs_new_protected:Nn __enumext_after_stop_list:
1137 {
1138 \tl_use:c { l__enumext_after_stop_list_ __enumext_level: _tl }
1139 }
```

The function `\__enumext_after_args_exec:` executes the `{\code}` set by the `first` key after the end of the second argument of the list defining the `enumext` environment, just before the first occurrence of `\item`: `\list{\arg one}{\arg two}{\code}\item`.

```
1140 \cs_new_protected:Nn __enumext_after_args_exec:
1141 {
1142 \tl_use:c { l__enumext_after_list_args_ __enumext_level: _tl }
1143 }
```

(End of definition for `\__enumext_before_args_exec:` and others.)

### 13.20.2 Functions for before, after and first keys in keyans

Same implementation as the one used in the `enumext` environment.

```
__enumext_before_args_exec_v:
__enumext_before_keys_exec_v:
__enumext_after_stop_list_v:
__enumext_after_args_exec_v:
1144 \cs_new_protected:Nn __enumext_before_args_exec_v:
1145 {
1146 \tl_use:N \l__enumext_before_starred_key_v_tl
1147 }
1148 \cs_new_protected:Nn __enumext_before_keys_exec_v:
1149 {
1150 \tl_use:N \l__enumext_before_no_starred_key_v_tl
1151 }
1152 \cs_new_protected:Nn __enumext_after_stop_list_v:
1153 {
1154 \tl_use:N \l__enumext_after_stop_list_v_tl
1155 }
1156 \cs_new_protected:Nn __enumext_after_args_exec_v:
1157 {
1158 \tl_use:N \l__enumext_after_list_args_v_tl
1159 }
```

(End of definition for `\__enumext_before_args_exec_v:` and others.)

### 13.20.3 Functions for before, after and first keys in enumext\* and keyans\*

Same implementation as the one used in the `enumext` environment.

```
__enumext_before_args_exec_vii:
__enumext_before_keys_exec_vii
__enumext_after_stop_list_vii:
__enumext_after_args_exec_vii:
1160 \cs_new_protected:Nn __enumext_before_args_exec_vii:
1161 {
1162 \tl_use:N \l__enumext_before_starred_key_vii_tl
1163 }
1164 \cs_new_protected:Nn __enumext_before_args_exec_viii:
1165 {
1166 \tl_use:N \l__enumext_before_starred_key_viii_tl
1167 }
1168 \cs_new_protected:Nn __enumext_before_keys_exec_vii:
1169 {
1170 \tl_use:N \l__enumext_before_no_starred_key_vii_tl
1171 }
1172 \cs_new_protected:Nn __enumext_before_keys_exec_viii:
1173 {
1174 \tl_use:N \l__enumext_before_no_starred_key_viii_tl
1175 }
1176 \cs_new_protected:Nn __enumext_after_stop_list_vii:
1177 {
1178 \tl_use:N \l__enumext_after_stop_list_vii_tl
1179 }
1180 \cs_new_protected:Nn __enumext_after_stop_list_viii:
1181 {
1182 \tl_use:N \l__enumext_after_stop_list_viii_tl
1183 }
1184 \cs_new_protected:Nn __enumext_after_args_exec_vii:
1185 {
1186 \tl_use:N \l__enumext_after_list_args_vii_tl
1187 }
1188 \cs_new_protected:Nn __enumext_after_args_exec_viii:
1189 {
1190 \tl_use:N \l__enumext_after_list_args_viii_tl
1191 }
```

(End of definition for `\__enumext_before_args_exec_vii`: and others.)

### 13.21 Setting keys for multicols and minipage

The default value of the `columns-sep` key is handled by the state of the boolean variable `\l__enumext_columns_sep_X_bool` which is handled in the internal definition of the `enumext` and `keyans` environments. Define and set `mini-env`, `mini-sep`, `columns-sep` and `columns` keys for `enumext`, `enumext*`, `keyans` and `keyans*` environments.

```

1192 \cs_set_protected:Npn __enumext_tmp:nn #1 #2
1193 {
1194 \keys_define:nn { enumext / #1 }
1195 {
1196 mini-env .dim_set:c = { __enumext_minipage_right_#2_dim },
1197 mini-env .value_required:n = true,
1198 mini-sep .dim_set:c = { __enumext_minipage_hsep_#2_dim },
1199 mini-sep .initial:n = 0.3333em,
1200 mini-sep .value_required:n = true,
1201 columns-sep .dim_set:c = { __enumext_columns_sep_#2_dim },
1202 columns-sep .value_required:n = true,
1203 columns .int_set:c = { __enumext_columns_#2_int },
1204 columns .initial:n = 1,
1205 columns .value_required:n = true,
1206 }
1207 }
1208 \clist_map_inline:Nn \c__enumext_all_envs_clist { __enumext_tmp:nn #1 }

```

For `enumext*` and `keyans*` environments the situation is a bit different, the command `\miniright` is not available, so we will add the keys `mini-right` and `mini-right*` to implement support for `minipage` environment.

```

1209 \cs_set_protected:Npn __enumext_tmp:nn #1 #2
1210 {
1211 \keys_define:nn { enumext / #1 }
1212 {
1213 mini-right .tl_gset:c = { g__enumext_miniright_code_#2_tl },
1214 mini-right .value_required:n = true,
1215 mini-right* .code:n = {
1216 \bool_gset_true:c { g__enumext_minipage_center_#2_bool }
1217 \keys_set:nn { enumext / #1 } { mini-right = {##1} }
1218 },
1219 mini-right* .value_required:n = true,
1220 }
1221 }
1222 \clist_map_inline:nn { {enumext*}{vii}, {keyans*}{viii} } { __enumext_tmp:nn #1 }

```

(End of definition for `mini-env` and others.)

### 13.22 Adjustment of vertical spaces for multicols

When nesting a “*list environment*” inside the `multicols` environment, the values of the “*vertical spaces*” are lost, basically the `multicols` environment takes control over them. Graphically it can be seen like in the figure 7.



Figure 7: Representation of the vertical space in `multicols` for a nested level.

To keep the desired spaces *above* and *below* in the “*list environment*” (`\topsep + [\partopsep]`) it is necessary to “*adjust*” the spaces added by the `multicols` environment. The most appropriate option in this case is to use a “*context sensitive*” vertical space with `\addvspace`.

I should make it clear that the implementation here is a “*bit questionable*”. At first glance doing `\multicolsep=\topsep` seemed right, but the results were not always as expected. An almost *imperceptible* detail is that in some cases the `\itemsep` values of are “*stretched*”, possibly due to the use of `\raggedcolumns` and this affects the lower space when closing the environment, which is “*smaller*” than expected. My attempts to find the correct values using `\showoutput` and `\showboxdepth` absolutely failed.



### 13.22.1 Adjustment of vertical spaces for multicol in enumext

`\__enumext_multi_set_vskip:` The function `\__enumext_multi_set_vskip:` will take care of determining the “*adjusted spaces*” that we will apply “*above*” and “*below*” the `multicol` environment in `enumext`.

We will set the default values taking into account that  $\text{\TeX}$  is in  $\langle \textit{horizontal mode} \rangle$ , then we will make the settings for the  $\langle \textit{vertical mode} \rangle$  in which `\partopsep` comes into play.

Set the values of `\l__enumext_multicols_above_X_skip` and `\l__enumext_multicols_below_X_skip` equal to the value of `\topsep` in the *current level*.

```

1223 \cs_new_protected:Nn __enumext_multi_set_vskip:
1224 {
1225 \skip_set:cn { \l__enumext_multicols_above_ __enumext_level: _skip }
1226 {
1227 \skip_use:c { \l__enumext_topsep_ __enumext_level: _skip }
1228 }
1229 \skip_set:cn { \l__enumext_multicols_below_ __enumext_level: _skip }
1230 {
1231 \skip_use:c { \l__enumext_topsep_ __enumext_level: _skip }
1232 }
1233 __enumext_add_pre_parsep:
1234 }

```

(End of definition for `\__enumext_multi_set_vskip:.`)

`\__enumext_add_pre_parsep:` The function `\__enumext_add_pre_parsep:` “*adjusted*” the value of `\l__enumext_multicols_above_X_skip` detecting the value of `\parsep` from the previous level. This is necessary since `\parsep` from the previous level affects the *vertical spaces*.

```

1235 \cs_new_protected:Nn __enumext_add_pre_parsep:
1236 {
1237 \int_case:nn { \l__enumext_level_int }
1238 {
1239 { 2 }{
1240 \skip_if_eq:nnF { \l__enumext_parsep_i_skip } { \c_zero_skip }
1241 {
1242 \skip_add:Nn \l__enumext_multicols_above_ii_skip
1243 {
1244 \l__enumext_parsep_i_skip
1245 }
1246 }
1247 }
1248 { 3 }{
1249 \skip_if_eq:nnF { \l__enumext_parsep_ii_skip } { \c_zero_skip }
1250 {
1251 \skip_add:Nn \l__enumext_multicols_above_iii_skip
1252 {
1253 \l__enumext_parsep_ii_skip
1254 }
1255 }
1256 }
1257 { 4 }{
1258 \skip_if_eq:nnF { \l__enumext_parsep_iii_skip } { \c_zero_skip }
1259 {
1260 \skip_add:Nn \l__enumext_multicols_above_iv_skip
1261 {
1262 \l__enumext_parsep_iii_skip
1263 }
1264 }
1265 }
1266 }
1267 }

```

(End of definition for `\__enumext_add_pre_parsep:.`)

`\__enumext_multi_addvspace:` The function `\__enumext_multi_addvspace:` will apply the spaces set using `\addvspace` “*above*” the `multicol` environment in `enumext`, taking into account whether  $\text{\TeX}$  is in  $\langle \textit{horizontal mode} \rangle$  or  $\langle \textit{vertical mode} \rangle$ .

```

1268 \cs_new_protected:Nn __enumext_multi_addvspace:
1269 {
1270 __enumext_multi_set_vskip:
1271 \mode_if_vertical:T
1272 {

```

```

1273 \skip_add:cn { __enumext_multicols_above_ __enumext_level: _skip }
1274 {
1275 \skip_use:c { __enumext_partopsep_ __enumext_level: _skip }
1276 }
1277 \skip_add:cn { __enumext_multicols_below_ __enumext_level: _skip }
1278 {
1279 \skip_use:c { __enumext_partopsep_ __enumext_level: _skip }
1280 }
1281 }
1282 \par\nopagebreak
1283 \addvspace{ \skip_use:c { __enumext_multicols_above_ __enumext_level: _skip } }
1284 }

```

(End of definition for `\__enumext_multi_addvspace:`)

### 13.22.2 Adjustment of vertical spaces for multicols in keyans

`\__enumext_keyans_multi_set_vskip:`  
`\__enumext_keyans_multi_addvspace:`

The function `\__enumext_keyans_multi_set_vskip:` will take care of determining the “adjusted spaces” that we will apply “above” and “below” the `\multicols` environment in `keyans`. The implementation of this function is the same as the one used in `enumext`.

```

1285 \cs_new_protected:Nn __enumext_keyans_multi_set_vskip:
1286 {
1287 \skip_set:Nn __enumext_multicols_above_v_skip
1288 {
1289 __enumext_topsep_v_skip
1290 }
1291 \skip_set:Nn __enumext_multicols_below_v_skip
1292 {
1293 __enumext_topsep_v_skip
1294 }
1295 }
1296 \cs_new_protected:Nn __enumext_keyans_multi_addvspace:
1297 {
1298 __enumext_keyans_multi_set_vskip:
1299 \mode_if_vertical:T
1300 {
1301 \skip_add:Nn __enumext_multicols_above_v_skip
1302 {
1303 \skip_use:N __enumext_partopsep_v_skip
1304 }
1305 \skip_add:Nn __enumext_multicols_below_v_skip
1306 {
1307 \skip_use:N __enumext_partopsep_v_skip
1308 }
1309 }
1310 \par\nopagebreak
1311 \addvspace{ __enumext_multicols_above_v_skip }
1312 }

```

(End of definition for `\__enumext_keyans_multi_set_vskip:` and `\__enumext_keyans_multi_addvspace:`)

### 13.23 Adjustment of vertical spaces for minipage

When nesting a “list environment” within the `minipage` environment, the values of the “vertical spaces” are lost. Graphically it can be seen like in the figure 8.



Figure 8: Representation of the `minipage` spacing adjustment for a nested level.

Since we want to keep the “left” and “right” environments “aligned on top”, preserving the `\baselineskip` and keep the desired “spaces” (`\topsep` + `[\partopsep]`) it is necessary to “adjust” the “vertical spaces” for `minipage` environments.

Here there are several complications that we must circumvent, the `minipage` environment eliminates the “top” spaces, the `\multicols` environment can be nested in the `minipage` environment, the “top” and “bottom” spaces are affected when `\topsep=0pt` and to this is added the `\partopsep` parameter that comes into action according to whether  $\TeX$  is in *horizontal mode* or *vertical mode*. Depending on these cases, small adjustments must be made using `\vspace` and `\addvspace` to obtain the “desired vertical spacing”.

- Again I must make clear that the implementation here is a “*bit questionable*”, but hunting the spaces (*glue*) produced by the `minipage` environment is quite complicated, even more if `multicols` it is nested. The setting of the values was more “*trial and error*” (aprox to `\strutbox`), using the help of the `lua-visual-debug`[14] package, again my attempts to find the correct values using `\showoutput` and `\showboxdepth` absolutely failed.

### 13.23.1 Adjustment of vertical spaces for minipage in enumext

`\__enumext_minipage_set_skip:`  
`\__enumext_minipage_add_space:`

The function `\__enumext_minipage_set_skip:` will take care of determining the “*adjust*” spaces that we will apply “*above*” and “*below*” the `__enumext_mini_page` environment in `enumext`.

First we will set the value of `\l__enumext_minipage_right_skip` equal to `\topsep`, then we will see if  $\text{\TeX}$  is in *vertical mode* and we will add `\partopsep`, followed by that we set the value of `\l__enumext_minipage_after_skip`.

```

1313 \cs_new_protected:Nn __enumext_minipage_set_skip:
1314 {
1315 \skip_set:Nn \l__enumext_minipage_right_skip
1316 {
1317 \skip_use:c { \l__enumext_topsep_ __enumext_level: _skip }
1318 }
1319 \mode_if_vertical:T
1320 {
1321 \skip_add:Nn \l__enumext_minipage_right_skip
1322 {
1323 \skip_use:c { \l__enumext_partopsep_ __enumext_level: _skip }
1324 }
1325 }
1326 \skip_set_eq:NN \l__enumext_minipage_after_skip \l__enumext_minipage_right_skip

```

We will adjust the values `\l__enumext_multicols_above_X_skip` and `\l__enumext_multicols_below_X_skip` and call the function `\__enumext_pre_itemsep_skip:`.

```

1327 \skip_set_eq:cN
1328 { \l__enumext_multicols_above_ __enumext_level: _skip } \l__enumext_minipage_right_skip
1329 \skip_set_eq:cN
1330 { \l__enumext_multicols_below_ __enumext_level: _skip } \l__enumext_minipage_right_skip
1331 __enumext_pre_itemsep_skip:

```

If the environment `multicols` is active, we set `\topskip=0pt` and then we make `\multicolsep` have the same value as `\l__enumext_multicols_above_X_skip`.

```

1332 \int_compare:nNnT
1333 { \int_use:c { \l__enumext_columns_ __enumext_level: _int } } > { 1 }
1334 {
1335 \skip_zero:N \topskip
1336 \skip_set_eq:Nc \multicolsep { \l__enumext_multicols_above_ __enumext_level: _skip }
1337 }
1338 }

```

The function `\__enumext_minipage_add_space:` will apply the spaces on the “*left side*” using `\addvspace` “*above*” the `__enumext_mini_page` environment, taking into account whether  $\text{\TeX}$  is in *horizontal mode* or *vertical mode*. Here we use the plain  $\text{\TeX}$  macro `\nointerlineskip` to prevent baseline “*glue*” being added between the next pair of boxes in a *vertical list*. For the latter we will make some adjustments since the `\partopsep` parameter comes into play and this affects the *vertical spacing*.

```

1339 \cs_new_protected:Nn __enumext_minipage_add_space:
1340 {
1341 __enumext_minipage_set_skip:
1342 __enumext_unskip_unkern:
1343 \mode_if_vertical:TF
1344 {
1345 \nopagebreak\nointerlineskip
1346 }
1347 {
1348 \par\nopagebreak\nointerlineskip
1349 \skip_zero:c { \l__enumext_partopsep_ __enumext_level: _skip }
1350 }
1351 \int_compare:nNnTF
1352 { \int_use:c { \l__enumext_columns_ __enumext_level: _int } } > { 1 }
1353 {
1354 \addvspace{ 0.445\box_ht:N \strutbox }
1355 }
1356 {
1357 \addvspace{ 0.250\box_ht:N \strutbox }
1358 }
1359 }

```

(End of definition for `\__enumext_minipage_set_skip:` and `\__enumext_minipage_add_space:.`)

`\__enumext_pre_itemsep_skip:` The function `\__enumext_pre_itemsep_skip:` will adjust the spaces below the environment `minipage` and the environment `multicols` if it is nested in it, taking into account the value of `\itemsep` from the previous level.

```

1360 \cs_new_protected:Nn __enumext_pre_itemsep_skip:
1361 {
1362 \int_case:nn { __enumext_level_int }
1363 {
1364 { 2 }{
1365 \skip_if_eq:nnTF
1366 { __enumext_itemsep_i_skip } { __enumext_minipage_after_skip }
1367 {
1368 \skip_set:Nn __enumext_minipage_after_skip { 0.150\box_ht:N \strutbox }
1369 \skip_set:Nn __enumext_multicols_below_ii_skip { 0.350\box_ht:N \strutbox }
1370 }
1371 {
1372 \dim_compare:nNnT
1373 { __enumext_itemsep_i_skip } < { __enumext_minipage_after_skip }
1374 {
1375 \skip_sub:Nn
1376 __enumext_minipage_after_skip { __enumext_itemsep_i_skip }
1377 \skip_sub:Nn
1378 __enumext_multicols_below_ii_skip { __enumext_itemsep_i_skip }
1379 \skip_add:Nn
1380 __enumext_minipage_after_skip { 0.150\box_ht:N \strutbox }
1381 \skip_add:Nn
1382 __enumext_multicols_below_ii_skip { 0.350\box_ht:N \strutbox }
1383 }
1384 \dim_compare:nNnT
1385 { __enumext_itemsep_i_skip } > { __enumext_minipage_after_skip }
1386 {
1387 \skip_set:Nn __enumext_minipage_temp_skip
1388 {
1389 __enumext_itemsep_i_skip - __enumext_minipage_after_skip
1390 }
1391 \skip_sub:Nn
1392 __enumext_minipage_after_skip { __enumext_itemsep_i_skip }
1393 \skip_sub:Nn
1394 __enumext_multicols_below_ii_skip { __enumext_itemsep_i_skip }
1395 \skip_add:Nn
1396 __enumext_minipage_after_skip
1397 { 0.150\box_ht:N \strutbox + __enumext_minipage_temp_skip }
1398 \skip_add:Nn
1399 __enumext_multicols_below_ii_skip
1400 { 0.350\box_ht:N \strutbox + __enumext_minipage_temp_skip }
1401 }
1402 }
1403 }
1404 { 3 }{
1405 \skip_if_eq:nnTF
1406 { __enumext_itemsep_ii_skip } { \c_zero_skip }
1407 {
1408 \skip_set:Nn __enumext_minipage_after_skip { 0.150\box_ht:N \strutbox }
1409 \skip_set:Nn __enumext_multicols_below_iii_skip { 0.350\box_ht:N \strutbox }
1410 }
1411 {
1412 \dim_compare:nNnT
1413 { __enumext_itemsep_ii_skip } < { __enumext_minipage_after_skip }
1414 {
1415 \skip_sub:Nn
1416 __enumext_minipage_after_skip { __enumext_itemsep_ii_skip }
1417 \skip_sub:Nn
1418 __enumext_multicols_below_iii_skip { __enumext_itemsep_ii_skip }
1419 \skip_add:Nn
1420 __enumext_minipage_after_skip { 0.150\box_ht:N \strutbox }
1421 \skip_add:Nn
1422 __enumext_multicols_below_iii_skip { 0.350\box_ht:N \strutbox }
1423 }
1424 \dim_compare:nNnT
1425 { __enumext_itemsep_ii_skip } > { __enumext_minipage_after_skip }

```

```

1426 {
1427 \skip_set:Nn \l__enumext_minipage_temp_skip
1428 {
1429 \l__enumext_itemsep_ii_skip - \l__enumext_minipage_after_skip
1430 }
1431 \skip_sub:Nn
1432 \l__enumext_minipage_after_skip { \l__enumext_itemsep_ii_skip }
1433 \skip_sub:Nn
1434 \l__enumext_multicols_below_iii_skip { \l__enumext_itemsep_ii_skip }
1435 \skip_add:Nn
1436 \l__enumext_minipage_after_skip
1437 { 0.150\box_ht:N \strutbox + \l__enumext_minipage_temp_skip }
1438 \skip_add:Nn
1439 \l__enumext_multicols_below_iii_skip
1440 { 0.350\box_ht:N \strutbox + \l__enumext_minipage_temp_skip }
1441 }
1442 }
1443 }
1444 { 4 }{
1445 \skip_if_eq:nnTF { \l__enumext_itemsep_iii_skip } { \c_zero_skip }
1446 {
1447 \skip_set:Nn \l__enumext_minipage_after_skip { 0.150\box_ht:N \strutbox }
1448 \skip_set:Nn \l__enumext_multicols_below_iv_skip { 0.350\box_ht:N \strutbox }
1449 }
1450 {
1451 \dim_compare:nNnT
1452 { \l__enumext_itemsep_iii_skip } < { \l__enumext_minipage_after_skip }
1453 {
1454 \skip_sub:Nn
1455 \l__enumext_minipage_after_skip { \l__enumext_itemsep_iii_skip }
1456 \skip_sub:Nn
1457 \l__enumext_multicols_below_iv_skip { \l__enumext_itemsep_iii_skip }
1458 \skip_add:Nn
1459 \l__enumext_minipage_after_skip { 0.150\box_ht:N \strutbox }
1460 \skip_add:Nn
1461 \l__enumext_multicols_below_iv_skip { 0.350\box_ht:N \strutbox }
1462 }
1463 \dim_compare:nNnT
1464 { \l__enumext_itemsep_iii_skip } > { \l__enumext_minipage_after_skip }
1465 {
1466 \skip_set:Nn \l__enumext_minipage_temp_skip
1467 {
1468 \l__enumext_itemsep_iii_skip - \l__enumext_minipage_after_skip
1469 }
1470 \skip_sub:Nn
1471 \l__enumext_minipage_after_skip { \l__enumext_itemsep_iii_skip }
1472 \skip_sub:Nn
1473 \l__enumext_multicols_below_iv_skip { \l__enumext_itemsep_iii_skip }
1474 \skip_add:Nn
1475 \l__enumext_minipage_after_skip
1476 { 0.150\box_ht:N \strutbox + \l__enumext_minipage_temp_skip }
1477 \skip_add:Nn
1478 \l__enumext_multicols_below_iv_skip
1479 { 0.350\box_ht:N \strutbox + \l__enumext_minipage_temp_skip }
1480 }
1481 }
1482 }
1483 }
1484 }

```

(End of definition for `\__enumext_pre_itemsep_skip:`)

### 13.23.2 Adjustment of vertical spaces for minipage in keyans

```

__enumext_keyans_minipage_set_skip:
__enumext_keyans_minipage_add_space:
__enumext_keyans_pre_itemsep_skip:

```

The function `\__enumext_keyans_mini_set_vskip:` will take care of determining the “adjusted” spaces that we will apply “above” and “below” the `__enumext_mini_page` environment in [keyans](#). The implementation of this function is the same as the one used in [enumext](#).

```

1485 \cs_new_protected:Nn __enumext_keyans_minipage_set_skip:
1486 {
1487 \skip_zero:N \l__enumext_minipage_after_skip
1488 \skip_zero:N \l__enumext_minipage_left_skip
1489 \skip_zero:N \l__enumext_minipage_right_skip

```

```

1490 \skip_set:Nn \l__enumext_minipage_right_skip
1491 {
1492 \l__enumext_topsep_v_skip
1493 }
1494 \mode_if_vertical:T
1495 {
1496 \skip_add:Nn \l__enumext_minipage_right_skip
1497 {
1498 \l__enumext_partopsep_v_skip
1499 }
1500 }
1501 \skip_set_eq:NN \l__enumext_minipage_after_skip \l__enumext_minipage_right_skip
1502 \skip_set_eq:NN \l__enumext_multicols_above_v_skip \l__enumext_minipage_right_skip
1503 \skip_set_eq:NN \l__enumext_multicols_below_v_skip \l__enumext_minipage_right_skip
1504 __enumext_keyans_pre_itemsep_skip:
1505 \int_compare:nNnT { \l__enumext_columns_v_int } > { 1 }
1506 {
1507 \skip_zero:N \topskip
1508 \skip_set_eq:NN \multicolsep \l__enumext_minipage_right_skip
1509 }
1510 }
1511 \cs_new_protected:Nn __enumext_keyans_minipage_add_space:
1512 {
1513 __enumext_keyans_minipage_set_skip:
1514 __enumext_unskip_unkern:
1515 \mode_if_vertical:TF
1516 {
1517 \nopagebreak\nointerlineskip
1518 }
1519 {
1520 \par\nopagebreak\nointerlineskip
1521 \skip_zero:N \l__enumext_partopsep_v_skip
1522 }
1523 \int_compare:nNnTF { \l__enumext_columns_v_int } > { 1 }
1524 {
1525 \addvspace{ 0.445\box_ht:N \strutbox }
1526 }
1527 {
1528 \addvspace{ 0.250\box_ht:N \strutbox }
1529 }
1530 }
1531 \cs_new_protected:Nn __enumext_keyans_pre_itemsep_skip:
1532 {
1533 \skip_if_eq:nnTF
1534 { \l__enumext_itemsep_i_skip } { \l__enumext_minipage_after_skip }
1535 {
1536 \skip_set:Nn \l__enumext_minipage_after_skip { 0.150\box_ht:N \strutbox }
1537 \skip_set:Nn \l__enumext_multicols_below_v_skip { 0.350\box_ht:N \strutbox }
1538 }
1539 {
1540 \dim_compare:nNnT
1541 { \l__enumext_itemsep_i_skip } < { \l__enumext_minipage_after_skip }
1542 {
1543 \skip_sub:Nn \l__enumext_minipage_after_skip { \l__enumext_itemsep_i_skip }
1544 \skip_sub:Nn \l__enumext_multicols_below_v_skip { \l__enumext_itemsep_i_skip }
1545 \skip_add:Nn \l__enumext_minipage_after_skip { 0.150\box_ht:N \strutbox }
1546 \skip_add:Nn \l__enumext_multicols_below_v_skip { 0.350\box_ht:N \strutbox }
1547 }
1548 \dim_compare:nNnT
1549 { \l__enumext_itemsep_i_skip } > { \l__enumext_minipage_after_skip }
1550 {
1551 \skip_set:Nn \l__enumext_minipage_temp_skip
1552 {
1553 \l__enumext_itemsep_i_skip - \l__enumext_minipage_after_skip
1554 }
1555 \skip_sub:Nn \l__enumext_minipage_after_skip { \l__enumext_itemsep_i_skip }
1556 \skip_sub:Nn \l__enumext_multicols_below_v_skip { \l__enumext_itemsep_i_skip }
1557 \skip_add:Nn \l__enumext_minipage_after_skip
1558 { 0.150\box_ht:N \strutbox + \l__enumext_minipage_temp_skip }
1559 \skip_add:Nn \l__enumext_multicols_below_v_skip
1560 { 0.350\box_ht:N \strutbox + \l__enumext_minipage_temp_skip }

```



```

1561 }
1562 }
1563 }

```

(End of definition for `\__enumext_keyans_minipage_set_skip:`, `\__enumext_keyans_minipage_add_space:`, and `\__enumext_keyans_pre_itemsep_skip:`.)

### 13.23.3 Adjustment of vertical spaces for minipage in enumext\* and keyans\*

The functions `\__enumext_mini_set_vskip_vii:` and `\__enumext_mini_set_vskip_viii:` will take care of determining the “adjusted” spaces that we will apply “above” and “below” the `\__enumext_mini_page` environment in `enumext*` and `keyans*`.

```

1564 \cs_new_protected:Nn __enumext_mini_set_vskip_vii:
1565 {
1566 \skip_zero_new:N \l__enumext_minipage_left_skip
1567 \skip_gzero_new:N \g__enumext_minipage_right_skip
1568 \skip_gzero_new:N \g__enumext_minipage_after_skip
1569 \skip_if_eq:nnTF { \l__enumext_topsep_vii_skip } { \c_zero_skip }
1570 {
1571 \skip_set:Nn \l__enumext_minipage_left_skip { 0.5\box_dp:N \strutbox }
1572 \skip_gset:Nn \g__enumext_minipage_right_skip { 0.325\box_dp:N \strutbox }
1573 }
1574 {
1575 \skip_set:Nn \l__enumext_minipage_left_skip { 0.5875\box_dp:N \strutbox }
1576 \skip_gset:Nn \g__enumext_minipage_right_skip
1577 {
1578 \l__enumext_topsep_vii_skip
1579 }
1580 \skip_gset:Nn \g__enumext_minipage_after_skip
1581 {
1582 0.325\box_dp:N \strutbox + \l__enumext_topsep_vii_skip
1583 }
1584 }
1585 }
1586 \cs_new_protected:Nn __enumext_mini_set_vskip_viii:
1587 {
1588 \skip_zero_new:N \l__enumext_minipage_after_skip
1589 \skip_zero_new:N \l__enumext_minipage_left_skip
1590 \skip_zero_new:N \l__enumext_minipage_right_skip
1591 \skip_if_eq:nnTF { \l__enumext_topsep_viii_skip } { \c_zero_skip }
1592 {
1593 \skip_set:Nn \l__enumext_minipage_left_skip
1594 {
1595 0.5\box_dp:N \strutbox
1596 }
1597 \skip_set:Nn \l__enumext_minipage_right_skip
1598 {
1599 \l__enumext_partopsep_viii_skip
1600 }
1601 \skip_set:Nn \l__enumext_minipage_after_skip
1602 {
1603 1.6\box_dp:N \strutbox
1604 }
1605 }
1606 {
1607 \skip_set:Nn \l__enumext_minipage_left_skip
1608 {
1609 0.5875\box_dp:N \strutbox
1610 }
1611 \skip_set:Nn \l__enumext_minipage_right_skip
1612 {
1613 \l__enumext_topsep_viii_skip
1614 }
1615 \skip_set:Nn \l__enumext_minipage_after_skip
1616 {
1617 0.325\box_dp:N \strutbox + \l__enumext_topsep_viii_skip
1618 }
1619 }
1620 }

```

(End of definition for `\__enumext_mini_set_vskip_vii:` and `\__enumext_mini_set_vskip_viii:`.)

`\__enumext_mini_addvspace_vii:`  
`\__enumext_mini_addvspace_viii:`

The functions `\__enumext_mini_addvspace_vii:` and `\__enumext_mini_addvspace_viii:` will apply the vertical space “only above” the `\__enumext_mini_page` environment on the *left side* when the `mini-right` key is active in the `enumext*` and `keyans*` environments. Here we will NOT take into account whether  $\TeX$  is in *(horizontal mode)* or *(vertical mode)*, since `\partopsep` is equal to `0pt` in both environments.

```

1621 \cs_new_protected:Nn __enumext_mini_addvspace_vii:
1622 {
1623 __enumext_mini_set_vskip_vii:
1624 \par\nopagebreak
1625 \addvspace { \l__enumext_minipage_left_skip }
1626 }
1627 \cs_new_protected:Nn __enumext_mini_addvspace_viii:
1628 {
1629 __enumext_mini_set_vskip_viii:
1630 \par\nopagebreak
1631 \addvspace { \l__enumext_minipage_left_skip }
1632 }

```

(End of definition for `\__enumext_mini_addvspace_vii:` and `\__enumext_mini_addvspace_viii:`.)

### 13.23.4 The command `\miniright`

The command `\miniright` will close the `\__enumext_mini_page` environment on the “left side”, open the `\__enumext_mini_page` environment on the “right side” adding the *adjusted vertical space*. By default we will add `\centering` when starting the “right side” environment. The *starred argument* ‘\*’ inhibits the use of `\centering` command i.e. the usual  $\TeX$  justification is maintained in the `\__enumext_mini_page` on the “right side”.

`\miniright`

First we will perform some checks to prevent the command from being executed outside the `enumext` environment or somewhere inappropriate then we will call the internal functions to execute it in the `enumext` and `keyans` environments.

```

1633 \NewDocumentCommand \miniright { s }
1634 {
1635 \int_compare:nNt { \l__enumext_keyans_pic_level_int } = { 1 }
1636 {
1637 \msg_error:nnn { enumext } { wrong-miniright-place }
1638 }
1639 % outside
1640 \bool_lazy_and:nnT
1641 { \int_compare_p:nNn { \l__enumext_level_int } = { 0 } }
1642 { \int_compare_p:nNn { \l__enumext_level_h_int } = { 0 } }
1643 {
1644 \msg_error:nnn { enumext } { wrong-miniright-place }
1645 }
1646 % starred env
1647 \bool_lazy_and:nnT
1648 { \bool_if_p:N \g__enumext_starred_bool }
1649 { \bool_not_p:n { \l__enumext_standar_bool } }
1650 {
1651 \msg_error:nnn { enumext } { wrong-miniright-starred }
1652 }
1653 % exec
1654 \int_compare:nNtF { \l__enumext_keyans_level_int } = { 1 }
1655 {
1656 __enumext_keyans_mini_right_cmd:n {#1}
1657 }
1658 { __enumext_mini_right_cmd:n {#1} }
1659 }

```

(End of definition for `\miniright`. This function is documented on page 11.)

`\__enumext_mini_right_cmd:n`

The function `\__enumext_mini_right_cmd:n` takes as argument the *starred* ‘\*’ of the `\miniright` command in the `enumext` environment. We check if the `mini-env` key is active via the variable `\l__enumext_minipage_right_X_dim`, if so we close the `multicols` environment with the `\__enumext_mini_page` environment on the “left side”, then we open the `\__enumext_mini_page` environment on the “right side”, apply our adjusted “vertical spaces”, followed by adding the `\centering` command when the *starred argument* ‘\*’ is not present and set zero `\g__enumext_minipage_stat_int`, otherwise we return an error.

```

1660 \cs_new_protected:Npn __enumext_mini_right_cmd:n #1
1661 {
1662 \dim_compare:nNtF
1663 { \dim_use:c { \l__enumext_minipage_right_ __enumext_level: _dim } } > { \c_zero_dim }

```

```

1664 {
1665 __enumext_multicols_stop:
1666 \int_compare:nNnT
1667 { \int_use:c { __enumext_columns_ __enumext_level: _int } } = { 1 }
1668 {
1669 \par\addvspace{ __enumext_minipage_after_skip }
1670 }
1671 \end__enumext_mini_page
1672 \hfill
1673 __enumext_mini_page{ \dim_use:c { __enumext_minipage_right_ __enumext_level: _dim } }
1674 \par\nointerlineskip
1675 \addvspace { __enumext_minipage_right_skip }
1676 \bool_if:nF {#1}
1677 {
1678 \centering
1679 }
1680 \int_gzero:N \g__enumext_minipage_stat_int
1681 }
1682 { \msg_error:nnn { enumext } { wrong-miniright-use } }
1683 % paranoia
1684 \RenewDocumentCommand \miniright { s }
1685 {
1686 \msg_error:nn { enumext } { many-miniright-used }
1687 }
1688 }

```

(End of definition for \\_\_enumext\_mini\_right\_cmd:n.)

\\_\_enumext\_keyans\_mini\_right\_cmd:n

The function \\_\_enumext\_keyans\_mini\_right\_cmd:n takes as argument the *starred* ‘\*’ of the \miniright command in the **keyans** environment. The implementation of this function is the same as that of the \\_\_enumext\_mini\_right\_cmd:n function of the **enumext** environment.

```

1689 \cs_new_protected:Npn __enumext_keyans_mini_right_cmd:n #1
1690 {
1691 \dim_compare:nNnTF { __enumext_minipage_right_v_dim } > { \c_zero_dim }
1692 {
1693 __enumext_keyans_multicols_stop:
1694 \int_compare:nNnT { __enumext_columns_v_int } = { 1 }
1695 {
1696 \par\addvspace{ __enumext_minipage_after_skip }
1697 }
1698 \end__enumext_mini_page
1699 \hfill
1700 __enumext_mini_page{ __enumext_minipage_right_v_dim }
1701 \par\nointerlineskip
1702 \addvspace { __enumext_minipage_right_skip }
1703 \bool_if:nF {#1}
1704 {
1705 \centering
1706 }
1707 \int_gzero:N \g__enumext_minipage_stat_int
1708 }
1709 { \msg_error:nnn { enumext } { wrong-miniright-use } }
1710 % paranoia
1711 \RenewDocumentCommand \miniright { s }
1712 {
1713 \msg_error:nn { enumext } { many-miniright-used }
1714 }
1715 }

```

(End of definition for \\_\_enumext\_keyans\_mini\_right\_cmd:n.)

### 13.24 Setting above and below keys

While having controlled the *vertical spaces* within the **enumext** and **keyans** environments when using the **columns** or **mini-env** keys, sometimes the “*vertical spaces above*” or “*vertical spaces below*” the environments are not as expected and it is necessary to be able to apply a “*fine correction*” to these. As I have not been able to correct these *glitches*, the best option is to leave a couple of *⟨keys⟩* dedicated to this purpose, in this case it is best to use **\vspace** or **\vspace\*** when convenient.

above Define above, above\*, below and below\* keys for **enumext** and **keyans** environments.

```

above* 1716 \cs_set_protected:Npn __enumext_tmp:nn #1 #2
below
below*

```

```

1717 {
1718 \keys_define:nn { enumext / #1 }
1719 {
1720 above .skip_set:c = { __enumext_vspace_above_#2_skip },
1721 above .value_required:n = true,
1722 above* .code:n = \bool_set_true:c { __enumext_vspace_a_star_#2_bool }
1723 \keys_set:nn { enumext / #1 } { above = {##1} },
1724 above* .value_required:n = true,
1725 below .skip_set:c = { __enumext_vspace_below_#2_skip },
1726 below .value_required:n = true,
1727 below* .code:n = \bool_set_true:c { __enumext_vspace_b_star_#2_bool }
1728 \keys_set:nn { enumext / #1 } { below = {##1} },
1729 below* .value_required:n = true,
1730 }
1731 }
1732 \clist_map_inline:Nn \c__enumext_all_envs_clist { __enumext_tmp:nn #1 }

```

(End of definition for *above* and *others*.)

### 13.24.1 Functions for *above* and *below* keys in *enumext*

`\__enumext_vspace_above:` The function `\__enumext_vspace_above:` apply the *vertical space above* the *enumext* environment set by the *above\** and *above* keys.

```

1733 \cs_new_protected:Nn __enumext_vspace_above:
1734 {
1735 \skip_if_eq:nnF
1736 { \skip_use:c { __enumext_vspace_above_ __enumext_level: _skip } } { \c_zero_skip }
1737 {
1738 \bool_if:cTF { __enumext_vspace_a_star_ __enumext_level: _bool }
1739 {
1740 \vspace*{ \skip_use:c { __enumext_vspace_above_ __enumext_level: _skip } }
1741 }
1742 {
1743 \vspace { \skip_use:c { __enumext_vspace_above_ __enumext_level: _skip } }
1744 }
1745 }
1746 }

```

(End of definition for `\__enumext_vspace_above:`.)

`\__enumext_vspace_below:` The function `\__enumext_vspace_below:` apply the *vertical space below* the *enumext* environment set by the *below\** and *below* keys.

```

1747 \cs_new_protected:Nn __enumext_vspace_below:
1748 {
1749 \skip_if_eq:nnF
1750 { \skip_use:c { __enumext_vspace_below_ __enumext_level: _skip } } { \c_zero_skip }
1751 {
1752 \bool_if:cTF { __enumext_vspace_b_star_ __enumext_level: _bool }
1753 {
1754 \vspace*{ \skip_use:c { __enumext_vspace_below_ __enumext_level: _skip } }
1755 }
1756 {
1757 \vspace { \skip_use:c { __enumext_vspace_below_ __enumext_level: _skip } }
1758 }
1759 }
1760 }

```

(End of definition for `\__enumext_vspace_below:`.)

### 13.24.2 Functions for *above* and *below* keys in *keyans*

`\__enumext_vspace_above_v:` The function `\__enumext_vspace_above_v:` apply the *vertical space above* the *keyans* environment set by the *above* and *above\** keys.

```

1761 \cs_new_protected:Nn __enumext_vspace_above_v:
1762 {
1763 \skip_if_eq:nnF { __enumext_vspace_above_v_skip } { \c_zero_skip }
1764 {
1765 \bool_if:NTF __enumext_vspace_a_star_v_bool
1766 {
1767 \vspace*{ __enumext_vspace_above_v_skip }
1768 }
1769 { \vspace { __enumext_vspace_above_v_skip } }
1770 }
1771 }

```

(End of definition for `\__enumext_vspace_above_v:`)

`\__enumext_vspace_below_v:`

The function `\__enumext_vspace_below_v:` apply the *vertical space below* the `keyans` environment set by the `below*` and `below` keys.

```
1772 \cs_new_protected:Nn __enumext_vspace_below_v:
1773 {
1774 \skip_if_eq:nnF { \l__enumext_vspace_below_v_skip } { \c_zero_skip }
1775 {
1776 \bool_if:NTF \l__enumext_vspace_b_star_v_bool
1777 {
1778 \vspace*{ \l__enumext_vspace_below_v_skip }
1779 }
1780 { \vspace { \l__enumext_vspace_below_v_skip } }
1781 }
1782 }
```

(End of definition for `\__enumext_vspace_below_v:`)

### 13.24.3 Functions for above and below keys in enumext\* keyans\*

`\__enumext_vspace_above_vii:`

The functions `\__enumext_vspace_above_vii:` and `\__enumext_vspace_above_viii:` apply the *vertical space above* the `enumext*` and `keyans*` environments set by the `above` and `above*` keys.

`\__enumext_vspace_above_viii:`

```
1783 \cs_new_protected:Nn __enumext_vspace_above_vii:
1784 {
1785 \skip_if_eq:nnF { \l__enumext_vspace_above_vii_skip } { \c_zero_skip }
1786 {
1787 \bool_if:NTF \l__enumext_vspace_a_star_vii_bool
1788 {
1789 \vspace*{ \l__enumext_vspace_above_vii_skip }
1790 }
1791 { \vspace { \l__enumext_vspace_above_vii_skip } }
1792 }
1793 }
1794 \cs_new_protected:Nn __enumext_vspace_above_viii:
1795 {
1796 \skip_if_eq:nnF { \l__enumext_vspace_above_viii_skip } { \c_zero_skip }
1797 {
1798 \bool_if:NTF \l__enumext_vspace_a_star_viii_bool
1799 {
1800 \vspace*{ \l__enumext_vspace_above_viii_skip }
1801 }
1802 { \vspace { \l__enumext_vspace_above_viii_skip } }
1803 }
1804 }
```

(End of definition for `\__enumext_vspace_above_vii:` and `\__enumext_vspace_above_viii:`)

`\__enumext_vspace_below_vii:`

The functions `\__enumext_vspace_below_vii:` and `\__enumext_vspace_below_viii:` apply the *vertical space below* the `enumext*` and `keyans*` environments set by the `below*` and `below` keys.

`\__enumext_vspace_below_viii:`

```
1805 \cs_new_protected:Nn __enumext_vspace_below_vii:
1806 {
1807 \skip_if_eq:nnF { \l__enumext_vspace_below_vii_skip } { \c_zero_skip }
1808 {
1809 \bool_if:NTF \l__enumext_vspace_b_star_vii_bool
1810 {
1811 \vspace*{ \l__enumext_vspace_below_vii_skip }
1812 }
1813 { \vspace { \l__enumext_vspace_below_vii_skip } }
1814 }
1815 }
1816 \cs_new_protected:Nn __enumext_vspace_below_viii:
1817 {
1818 \skip_if_eq:nnF { \l__enumext_vspace_below_viii_skip } { \c_zero_skip }
1819 {
1820 \bool_if:NTF \l__enumext_vspace_b_star_viii_bool
1821 {
1822 \vspace*{ \l__enumext_vspace_below_viii_skip }
1823 }
1824 { \vspace { \l__enumext_vspace_below_viii_skip } }
1825 }
1826 }
```

(End of definition for `\__enumext_vspace_below_vii:` and `\__enumext_vspace_below_viii:`)

### 13.25 Setting series, resume and resume\* keys

The `series` key is responsible for the whole process of the `resume` and `resume*` keys. The idea behind this is to be able to absorb the `<keys>` passed to the *optional argument* of the “first level” of the environments `enumext` and `enumext*`, but, discarding some specific `<keys>`. This implementation is adapted directly from the code provided by Jonathan P. Spratte (@Skillmon) in `chat-Tex-SX`

```

series We define the keys series, resume and resume* only for the “first level” of enumext and enumext*.
resume
resume*
1827 \cs_set_protected:Npn __enumext_tmp:n #1
1828 {
1829 \keys_define:nn { enumext / #1 }
1830 {
1831 series .str_set:N = __enumext_series_str,
1832 series .value_required:n = true,
1833 resume .code:n = __enumext_resume_series:n {##1},
1834 resume* .code:n = __enumext_resume_starred:,
1835 resume* .value_forbidden:n = true,
1836 }
1837 }
1838 \clist_map_inline:nn { level-1, enumext* } { __enumext_tmp:n {#1} }

```

(End of definition for series, resume, and resume\*.)

#### 13.25.1 Internal functions for series key

The function `\__enumext_filter_series:n` will be in charge of filtering the `<keys>` we want to store where `{#1}` represents the *optional argument* passed to the environment.

```

__enumext_filter_series:n
 __enumext_filter_series_key:n
 __enumext_filter_series_pair:nn
1839 \cs_new:Npn __enumext_filter_series:n #1
1840 {
1841 \use:e
1842 {
1843 \keyval_parse:NNn
1844 __enumext_filter_series_key:n
1845 __enumext_filter_series_pair:nn {#1}
1846 }
1847 }

```

The function `\__enumext_filter_series_key:n` will be responsible for filtering the `<keys>` that are passed “without value” by excluding the `resume`, `resume*` and `base-fix` keys.

```

1848 \cs_new:Npn __enumext_filter_series_key:n #1
1849 {
1850 \str_case:nnF {#1}
1851 {
1852 { resume } {} { resume* } {} { base-fix } {}
1853 }
1854 { , { \exp_not:n {#1} } }
1855 }

```

The function `\__enumext_filter_series_pair:nn` will be responsible for filtering the `<keys>` that are passed “with value” by excluding the `series`, `resume`, `start`, `start*`, `save-ans` and `save-key` keys.

```

1856 \cs_new:Npn __enumext_filter_series_pair:nn #1#2
1857 {
1858 \str_case:nnF {#1}
1859 {
1860 { series } {} { resume } {} { start } {}
1861 { start* } {} { save-ans } {} { save-key } {}
1862 }
1863 { , { \exp_not:n {#1} } = { \exp_not:n {#2} } }
1864 }

```

(End of definition for `\__enumext_filter_series:n`, `\__enumext_filter_series_key:n`, and `\__enumext_filter_series_pair:nn`.)

The function `\__enumext_parse_series:n` will be responsible for storing the filtered `<keys>` in the global variable `\g__enumext_series_<series name>_tl` along with the creation of the integer variable `\g__enumext_series_<series name>_int` when the key is passed as an argument; otherwise, it will check the state of the boolean variable `\l__enumext_resume_active_bool` set by the keys `resume` and `resume*` and will call the function `\__enumext_resume_last:n`.

- The value of boolean variable `\l__enumext_resume_active_bool` is set to true by the function `\__enumext_resume_counter:n` which is used by the keys `resume` and `resume*`, in this case we must Make sure it is set to false so that it does not overwrite the default filtered `<keys>`. This function is passed to the function `\__enumext_parse_keys:n` in the `enumext` environment definition (§13.39) and to the function `\__enumext_parse_keys_vii:n` in the `enumext*` environment definition (§13.44).



```

1865 \cs_new_protected:Npn __enumext_parse_series:n #1
1866 {
1867 \str_if_empty:NTF \l__enumext_series_str
1868 {
1869 \bool_if:NF \l__enumext_resume_active_bool
1870 {
1871 __enumext_resume_last:n {#1}
1872 }
1873 }
1874 {
1875 \tl_gclear_new:c { g__enumext_series_ \l__enumext_series_str _tl }
1876 \tl_gset:ce { g__enumext_series_ \l__enumext_series_str _tl }
1877 { __enumext_filter_series:n {#1} }
1878 \int_if_exist:cF { g__enumext_series_ \l__enumext_series_str _int }
1879 {
1880 \int_new:c { g__enumext_series_ \l__enumext_series_str _int }
1881 }
1882 }
1883 }

```

The function `\__enumext_resume_last:n` will be in charge of saving the filtering (*keys*) when the *series* key is *not used* and will save them in the variable `\g__enumext_standar_series_tl` for the *enumext* environment and in the variable `\g__enumext_starred_series_tl` for the *enumext\** environment.

```

1884 \cs_new_protected:Npn __enumext_resume_last:n #1
1885 {
1886 \bool_if:NT \l__enumext_standar_first_bool
1887 {
1888 \tl_gclear:N \g__enumext_standar_series_tl
1889 \tl_gset:Ne \g__enumext_standar_series_tl { __enumext_filter_series:n {#1} }
1890 }
1891 \bool_if:NT \l__enumext_starred_first_bool
1892 {
1893 \tl_gclear:N \g__enumext_starred_series_tl
1894 \tl_gset:Ne \g__enumext_starred_series_tl { __enumext_filter_series:n {#1} }
1895 }
1896 }

```

(End of definition for `\__enumext_parse_series:n` and `\__enumext_resume_last:n`.)

### 13.25.2 Internal function to save counter value

`\__enumext_resume_save_counter:`

The `\__enumext_resume_save_counter:` function will save the last counter value to `\g__enumext_series_⟨series name⟩_int` if the *series*={⟨series name⟩} key has been passed, to `\g__enumext_resume_int` if it has passed the key *resume without value* and the key *series* is not active, in `\g__enumext_series_⟨series name⟩_int` if the key *resume*={⟨series name⟩} has been passed and in `\g__enumext_series_⟨store name⟩_int` if the key has been passed *save-ans*={⟨store name⟩}.

- The variables `\l__enumext_series_str` and `\l__enumext__resume_name_tl` contain the same {⟨series name⟩} but are executed at different moments, the integer variable with `\l__enumext_series_str` sets the value when execute *series*={⟨series name⟩} and the integer variable with `\l__enumext__resume_name_tl` sets the subsequent values when use *resume*={⟨series name⟩}. This function is passed to the *enumext* environment definition (§13.39) and the *enumext\** environment definition (§13.44).

```

1897 \cs_new_protected:Npn __enumext_resume_save_counter:
1898 {
1899 \bool_if:NT \g__enumext_standar_bool
1900 {
1901 \tl_if_empty:NF \l__enumext_series_str
1902 {
1903 \int_gset_eq:cN
1904 { g__enumext_series_ \l__enumext_series_str _int } \value{enumXi}
1905 }
1906 \tl_if_empty:NTF \l__enumext_resume_name_tl
1907 {
1908 \str_if_empty:NF \l__enumext_series_str
1909 {
1910 \int_gset_eq:NN \g__enumext_resume_int \value{enumXi}
1911 }
1912 }
1913 {
1914 \int_if_exist:cT { g__enumext_series_ \l__enumext_resume_name_tl _int }
1915 {
1916 \int_gset_eq:cN

```

```

1917 { g__enumext_series_ \l__enumext_resume_name_tl _int } \value{enumXi}
1918 }
1919 }
1920 \int_if_exist:cT { g__enumext_resume_ \l__enumext_store_name_tl _int }
1921 {
1922 \int_gset_eq:cN
1923 { g__enumext_resume_ \l__enumext_store_name_tl _int } \value{enumXi}
1924 }
1925 }
1926 \bool_if:NT \g__enumext_starred_bool
1927 {
1928 \tl_if_empty:NF \l__enumext_series_str
1929 {
1930 \int_gset_eq:cN
1931 { g__enumext_series_ \l__enumext_series_str _int } \value{enumXvii}
1932 }
1933 \tl_if_empty:NTF \l__enumext_resume_name_tl
1934 {
1935 \str_if_empty:NT \l__enumext_series_str
1936 {
1937 \int_gset_eq:NN \g__enumext_resume_vii_int \value{enumXvii}
1938 }
1939 }
1940 {
1941 \int_if_exist:cT { g__enumext_series_ \l__enumext_resume_name_tl _int }
1942 {
1943 \int_gset_eq:cN
1944 { g__enumext_series_ \l__enumext_resume_name_tl _int } \value{enumXvii}
1945 }
1946 }
1947 \int_if_exist:cT { g__enumext_resume_ \l__enumext_store_name_tl _int }
1948 {
1949 \int_gset_eq:cN
1950 { g__enumext_resume_ \l__enumext_store_name_tl _int } \value{enumXvii}
1951 }
1952 }
1953 }

```

(End of definition for \\_\_enumext\_resume\_save\_counter:.)

### 13.25.3 Internal functions for resume key

\\_\_enumext\_resume\_series:n

The function \\_\_enumext\_resume\_series:n will handle the argument passed to the `resume` key in `enumext` and `enumext*` environments. If the key is passed *without value* the function \\_\_enumext\_resume\_counter: is executed which will set the counter according to the numbering of the last `enumext` or `enumext*` environments in which `series={⟨series name⟩}` key is not present, if the `save-ans` key is active it will set the counter according to the value of the integer variable created by that key, otherwise it will verify that the `\g__enumext_series_⟨series name⟩_tl` variable set by the `series` key exists, if so it will pass these keys to the *first level* of the environment, otherwise it will return an error.

```

1954 \cs_new_protected:Npn __enumext_resume_series:n #1
1955 {
1956 \tl_if_empty:NTF {#1}
1957 {
1958 __enumext_resume_counter:n { }
1959 }
1960 {
1961 \tl_if_exist:cTF { g__enumext_series_ \tl_to_str:n {#1} _tl }
1962 {
1963 __enumext_resume_counter:n {#1}
1964 \bool_if:NT \g__enumext_standar_bool
1965 {
1966 \keys_set:nv { enumext / level-1 }
1967 { g__enumext_series_ \tl_to_str:n {#1} _tl }
1968 }
1969 \bool_if:NT \g__enumext_starred_bool
1970 {
1971 \keys_set:nv { enumext / enumext* }
1972 { g__enumext_series_ \tl_to_str:n {#1} _tl }
1973 }
1974 }
1975 {
1976 \bool_if:NT \g__enumext_standar_bool

```

```

1977 {
1978 \msg_error:nnn { enumext } { unknown-series } {#1}
1979 }
1980 \bool_if:NT \g__enumext_starred_bool
1981 {
1982 \msg_error:nnn { enumext } { unknown-series } {#1}
1983 }
1984 }
1985 }
1986 }

```

(End of definition for `\__enumext_resume_series:n`)

```

__enumext_resume_counter:n
__enumext_resume_counter:
 __enumext_resume_counter_series:
 __enumext_resume_counter_save_ans:

```

The function `\__enumext_resume_counter:n` will set the variable `\l__enumext_resume_active_bool` to true and pass the value of the key `resume` to the variable `\l__enumext_series_name_tl` which will contain the `{\series name}`. If the variable `\l__enumext_series_name_tl` is empty, that is, we are passing the key `resume` *without value*, we will execute the function `\__enumext_resume_counter:`; otherwise, when we pass `resume={\series name}` we will execute the function `\__enumext_resume_counter_series:`, finally we will execute the function `\__enumext_resume_counter_save_ans:` which is associated with the key `save-ans`.

```

1987 \cs_new_protected:Npn __enumext_resume_counter:n #1
1988 {
1989 \bool_set_true:N \l__enumext_resume_active_bool
1990 \tl_set:Nn \l__enumext_resume_name_tl {#1}
1991 \tl_if_empty:NTF \l__enumext_resume_name_tl
1992 {
1993 __enumext_resume_counter:
1994 }
1995 {
1996 __enumext_resume_counter_series:
1997 }
1998 __enumext_resume_counter_save_ans:
1999 }

```

The `\__enumext_resume_counter:` function is executed when the `resume` key is used *without value*, only the counters for the “first level” of the environments will be set.

```

2000 \cs_new_protected:Nn __enumext_resume_counter:
2001 {
2002 \bool_if:NT \g__enumext_standar_bool
2003 {
2004 \int_gincr:N \g__enumext_resume_int
2005 \int_set_eq:NN \l__enumext_start_i_int \g__enumext_resume_int
2006 }
2007 \bool_if:NT \g__enumext_starred_bool
2008 {
2009 \int_gincr:N \g__enumext_resume_vii_int
2010 \int_set_eq:NN \l__enumext_start_vii_int \g__enumext_resume_vii_int
2011 }
2012 }

```

The function `\__enumext_resume_counter_series:` will be executed when the `resume={\series name}` key is active, setting the counters for the “first level” of the environments according to the value of the integer variables created by the `series` key.

```

2013 \cs_new_protected:Nn __enumext_resume_counter_series:
2014 {
2015 \bool_if:NT \g__enumext_standar_bool
2016 {
2017 \int_set:Nn \l__enumext_start_i_int
2018 {
2019 \int_use:c { g__enumext_series_ \l__enumext_resume_name_tl _int } + 1
2020 }
2021 }
2022 \bool_if:NT \g__enumext_starred_bool
2023 {
2024 \int_set:Nn \l__enumext_start_vii_int
2025 {
2026 \int_use:c { g__enumext_series_ \l__enumext_resume_name_tl _int } + 1
2027 }
2028 }
2029 }

```

The function `\__enumext_resume_counter_save_ans:` will be executed when the `save-ans` key is active along with the `resume` key, setting the counters for the “first level” of the environments according to the value of the integer variables created by the `save-ans` key.

```

2030 \cs_new_protected:Nn __enumext_resume_counter_save_ans:
2031 {
2032 \bool_lazy_and:nnT
2033 { \bool_if_p:N __enumext_standar_first_bool }
2034 { \bool_if_p:N __enumext_store_active_bool }
2035 {
2036 \int_set:Nn __enumext_start_i_int
2037 {
2038 \int_use:c { g__enumext_resume_ __enumext_store_name_tl _int } + 1
2039 }
2040 }
2041 \bool_lazy_and:nnT
2042 { \bool_if_p:N __enumext_starred_first_bool }
2043 { \bool_if_p:N __enumext_store_active_bool }
2044 {
2045 \int_set:Nn __enumext_start_vii_int
2046 {
2047 \int_use:c { g__enumext_resume_ __enumext_store_name_tl _int } + 1
2048 }
2049 }
2050 }

```

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

### 13.25.4 Internal function for `resume*` key

`\__enumext_resume_starred:` The function `\__enumext_resume_starred:` will handle the `resume*` key in the `enumext` and `enumext*` environments. This function will execute the filtered `<keys>` in the last one and will continue with the numbering according to the last execution of the environment `enumext` or `enumext*` in which the keys `resume={<series name>}` or `series={<series name>}` were not active.

```

2051 \cs_new_protected:Nn __enumext_resume_starred:
2052 {
2053 \bool_if:NT \g__enumext_standar_bool
2054 {
2055 \tl_if_empty:NF \g__enumext_standar_series_tl
2056 {
2057 __enumext_resume_counter:n { }
2058 \keys_set:nV { enumext / level-1 } \g__enumext_standar_series_tl
2059 }
2060 }
2061 \bool_if:NT \g__enumext_starred_bool
2062 {
2063 \tl_if_empty:NF \g__enumext_starred_series_tl
2064 {
2065 __enumext_resume_counter:n { }
2066 \keys_set:nV { enumext / enumext* } \g__enumext_starred_series_tl
2067 }
2068 }
2069 }

```

(End of definition for `\__enumext_resume_starred:`.)

## 13.26 Setting `save-ans`, `check-ans` and `no-store` keys

The key `save-ans` is directly associated with the keys `check-ans`, `no-store`, `resume` and `resume*`, this will activate the entire “storage system” in the `enumext` package.

### 13.26.1 Setting `save-ans` key

`save-ans` We define the keys `save-ans` only for the “first level” of `enumext` and `enumext*`.

```

2070 \cs_set_protected:Npn __enumext_tmp:n #1
2071 {
2072 \keys_define:nn { enumext / #1 }
2073 {
2074 save-ans .code:n = __enumext_storing_set:n {##1},
2075 save-ans .value_required:n = true,
2076 }
2077 }
2078 \clist_map_inline:nn { level-1, enumext* } { { __enumext_tmp:n {#1} } }

```

(End of definition for `save-ans`.)

### 13.26.2 Internal functions for save-ans key

```

__enumext_start_save_ans_msg:
__enumext_stop_save_ans_msg:

```

The functions `\__enumext_start_save_ans_msg:` and `\__enumext_stop_save_ans_msg:` will display in the terminal and `.log` file the environment in which the `save-ans` key was executed along with the line at the beginning and end of it. The function `\__enumext_start_save_ans_msg:` will be passed to `\__enumext_storing_set:n` and the function `\__enumext_stop_save_ans_msg:` will be passed to the function `\__enumext_execute_after_env:`.

```

2079 \cs_new_protected:Nn __enumext_start_save_ans_msg:
2080 {
2081 \msg_term:nnVV { enumext } { save-ans-log }
2082 \g__enumext_envir_name_tl \l__enumext_store_name_tl
2083 }
2084 \cs_new_protected:Nn __enumext_stop_save_ans_msg:
2085 {
2086 \msg_term:nnVV { enumext } { save-ans-log-hook }
2087 \g__enumext_envir_name_tl \g__enumext_store_name_tl
2088 }

```

(End of definition for `\__enumext_start_save_ans_msg:` and `\__enumext_stop_save_ans_msg:`.)

```

__enumext_storing_set:n
__enumext_storing_exec:

```

The function `\__enumext_storing_set:n` first pass the value of the `save-ans` key to the variable `\l__enumext_store_name_tl` which will contain the `{⟨store name⟩}` of the *sequence* and *prop list* we will use. If `\l__enumext_store_name_tl` is *empty* we return an error message, otherwise will return the appropriate message `\__enumext_start_save_ans_msg:` and proceed to execute the function `\__enumext_storing_exec:` for `enumext` and `enumext*` environments.

```

2089 \cs_new_protected:Npn __enumext_storing_set:n #1
2090 {
2091 \tl_set:Nx \l__enumext_store_name_tl {#1}
2092 \tl_if_empty:NTF \l__enumext_store_name_tl
2093 {
2094 \bool_lazy_or:nnT
2095 { \l__enumext_standar_first_bool } { \l__enumext_starred_first_bool }
2096 {
2097 \msg_error:nnV { enumext } { save-ans-empty } \g__enumext_envir_name_tl
2098 }
2099 }
2100 {
2101 \bool_lazy_or:nnT
2102 { \l__enumext_standar_first_bool } { \l__enumext_starred_first_bool }
2103 {
2104 __enumext_start_save_ans_msg:
2105 __enumext_storing_exec:
2106 }
2107 }
2108 }

```

The function `\__enumext_storing_exec:` will set to true the variable `\l__enumext_store_active_bool` which activates the use of the `\anskey` command and the `anskey*`, `keyans`, `keyans*` and `keyanspic` environments and will set to “true” the variable `\l__enumext_check_answers_bool` used for internal checking answers mechanism set by the `check-ans` and `no-store` keys, copy `{⟨store name⟩}` into the variable `\g__enumext_store_name_tl` and execute the function `\__enumext_anskey_env_make:V` creating the environment `anskey*` (§13.31).

```

2109 \cs_new_protected:Nn __enumext_storing_exec:
2110 {
2111 \bool_set_true:N \l__enumext_store_active_bool
2112 \bool_set_true:N \l__enumext_check_answers_bool
2113 \tl_gset:NV \g__enumext_store_name_tl \l__enumext_store_name_tl
2114 __enumext_anskey_env_make:V \l__enumext_store_name_tl

```

The *prop list* `\g__enumext_series_⟨store name⟩_prop` and the *sequence* `\g__enumext_series_⟨store name⟩_seq` will be created globally to “store content” in case they do not exist together with the integer variable `\g__enumext_series_⟨store name⟩_int` used by the keys `resume` and `resume*`.

```

2115 \prop_if_exist:cF { g__enumext_ \l__enumext_store_name_tl _prop }
2116 {
2117 \msg_log:nnV { enumext } { store-prop } \l__enumext_store_name_tl
2118 \prop_new:c { g__enumext_ \l__enumext_store_name_tl _prop }
2119 }
2120 \seq_if_exist:cF { g__enumext_ \l__enumext_store_name_tl _seq }
2121 {
2122 \msg_log:nnV { enumext } { store-seq } \l__enumext_store_name_tl
2123 \seq_new:c { g__enumext_ \l__enumext_store_name_tl _seq }

```

```

2124 }
2125 \int_if_exist:cF { g__enumext_resume_ \l__enumext_store_name_tl _int }
2126 {
2127 \msg_log:nnV { enumext } { store-int } \l__enumext_store_name_tl
2128 \int_new:c { g__enumext_resume_ \l__enumext_store_name_tl _int }
2129 }
2130 }

```

(End of definition for `\__enumext_storing_set:n` and `\__enumext_storing_exec:.`)

### 13.26.3 The check answer mechanism

The internal mechanism for “checking answers” follows this logic:

If the line begins with `\item` or `\item*` and does NOT *open a nested environment*, each `\item` or `\item*` must contain a *single* execution of the `\anskey` command, i.e. the counter of the executions of the `\anskey` command must be equal to the counter associated with the sum of executions of `\item` and `\item*`.

If the line begins with `\item` or `\item*` and *opens a nested environment* each `\item` or `\item*` in the nested environment must have a *single* execution of the `\anskey` command and the counter associated to the sum of `\item` and `\item*` executions must decrementing by “one” to maintain equality.

In order for the mechanism for the check-answer to work (not counting `keyans`, `keyans*` and `keyanspic`) we need:

1. We must keep track of the total number of `\item` and `\item*` (enumerated) that appear within the environment including the nested levels.
2. We must keep track of the total number of `\item` and `\item*` (enumerated) that appear per level of nesting.
3. Keeping track of the number of times the environment nests.

The integer variable associated to the sum of each `\item` and `\item*` in the environment `\g__enumext_item_number_int` must match the integer variable `\g__enumext_item_anskey_int` associated to the execution of the command `\anskey`. We analyze the cases:

- a) If the list only has one level the number of `\item` + `\item*` = `\anskey`
- b) If the list has *nested levels*, for each level of nesting we need to decrementing by one (for the `\item` or `\item*` that opens the nest) so that the account remains the same.

With `keyans`, `keyans*` and `keyanspic` it is enough to increase in one the integer of `\anskey`. The integers created must be global if they are not lost in the interior levels of nesting and to execute the test we will use a “hook” function after closing the *first level* of the environment.

### 13.26.4 Setting check-ans and no-store keys

Now we define the keys `check-ans` and `no-store` for all levels of `enumext` and `enumext*` environments.

```

check-ans 2131 \cs_set_protected:Npn __enumext_tmp:n #1
no-store 2132 {
2133 \keys_define:nn { enumext / #1 }
2134 {
2135 check-ans .bool_set:N = \l__enumext_check_ans_key_bool,
2136 check-ans .initial:n = false,
2137 check-ans .value_required:n = true,
2138 no-store .code:n = {
2139 \bool_set_false:N \l__enumext_check_answers_bool
2140 \bool_set_false:N \l__enumext_check_ans_key_bool
2141 },
2142 no-store .value_forbidden:n = true,
2143 }
2144 }
2145 \clist_map_inline:nn
2146 {
2147 level-1, level-2, level-3, level-4, enumext*
2148 }
2149 { __enumext_tmp:n {#1} }

```

(End of definition for `check-ans` and `no-store`.)

### 13.26.5 Set-up check answer mechanism

`\__enumext_check_ans_active:` The function `\__enumext_check_ans_active:` will first check the state of the variable `\l__enumext_store_name_tl`, that is, the `save-ans` key is active, if so it will check the state of the variable `\l__enumext_check_answers_bool` handled by the key `no-store` and will execute the function `\__enumext_check_ans_level:` only if “*true*”, i.e. the key `no-store` is not active.

```

2150 \cs_new_protected:Nn __enumext_check_ans_active:
2151 {
2152 \tl_if_empty:NF \l__enumext_store_name_tl
2153 {
2154 \bool_if:NT \l__enumext_check_answers_bool
2155 {
2156 __enumext_check_ans_level:
2157 }
2158 }
2159 }

```

The function `\__enumext_check_ans_level:` will decrement by “*one*” the value of the variable `\g__enumext_item_number_int` which keeps track of the executions of `\item` and `\item*` for each level of nesting of the environment `enumext`, taking into account whether it is nested within `enumext*` or the opposite and set `\l__enumext_item_number_bool` to “*false*”.

```

2160 \cs_new_protected:Nn __enumext_check_ans_level:
2161 {
2162 \int_case:nn { \l__enumext_level_int }
2163 {
2164 { 1 }{
2165 \bool_lazy_all:nT
2166 {
2167 { \bool_if_p:N \g__enumext_starred_bool }
2168 { \int_compare_p:nNn { \l__enumext_level_h_int } = { 1 } }
2169 }
2170 {
2171 \int_gdecr:N \g__enumext_item_number_int
2172 \bool_set_false:N \l__enumext_item_number_bool
2173 }
2174 }
2175 { 2 }{
2176 \int_gdecr:N \g__enumext_item_number_int
2177 \bool_set_false:N \l__enumext_item_number_bool
2178 }
2179 { 3 }{
2180 \int_gdecr:N \g__enumext_item_number_int
2181 \bool_set_false:N \l__enumext_item_number_bool
2182 }
2183 { 4 }{
2184 \int_gdecr:N \g__enumext_item_number_int
2185 \bool_set_false:N \l__enumext_item_number_bool
2186 }
2187 }

```

We should only execute this if `enumext*` is nested in the “*first level*” of `enumext`, for the rest of the cases the value of `\g__enumext_item_number_int` is already decreased.

```

2188 \int_case:nn { \l__enumext_level_h_int }
2189 {
2190 { 1 }{
2191 \bool_lazy_all:nT
2192 {
2193 { \bool_if_p:N \g__enumext_standar_bool }
2194 { \int_compare_p:nNn { \l__enumext_level_int } = { 1 } }
2195 }
2196 {
2197 \int_gdecr:N \g__enumext_item_number_int
2198 \bool_set_false:N \l__enumext_item_number_bool
2199 }
2200 }
2201 }
2202 }

```

(End of definition for `\__enumext_check_ans_active:` and `\__enumext_check_ans_level:`)



\\_\_enumext\_check\_ans\_key\_hook:

The function \\_\_enumext\_check\_ans\_key\_hook: will *export* the status of the local variable \l\_\_enumext\_check\_ans\_key\_bool to the global variable \g\_\_enumext\_check\_ans\_key\_bool only if the key `check-ans` is active.

```

2203 \cs_new_protected:Nn __enumext_check_ans_key_hook:
2204 {
2205 \bool_lazy_and:nnT
2206 { \bool_if_p:N \l__enumext_check_ans_key_bool }
2207 { \bool_if_p:N \g__enumext_standar_bool }
2208 {
2209 \bool_gset_true:N \g__enumext_check_ans_key_bool
2210 }
2211 \bool_lazy_and:nnT
2212 { \bool_if_p:N \l__enumext_check_ans_key_bool }
2213 { \bool_if_p:N \g__enumext_starred_bool }
2214 {
2215 \bool_gset_true:N \g__enumext_check_ans_key_bool
2216 }
2217 }

```

(End of definition for \\_\_enumext\_check\_ans\_key\_hook:.)

\\_\_enumext\_item\_answer\_diff:

The function \\_\_enumext\_item\_answer\_diff: will set the value of the variable \g\_\_enumext\_item\_answer\_diff\_int which is used by the functions \\_\_enumext\_check\_ans\_show: for the key `save-ans` and by the function \\_\_enumext\_check\_ans\_log: by the internal “*check answer*” mechanism. This function will be passed to the function \\_\_enumext\_execute\_after\_env:.

```

2218 \cs_new_protected:Nn __enumext_item_answer_diff:
2219 {
2220 \int_gset:Nn \g__enumext_item_answer_diff_int
2221 {
2222 \int_sign:n { \g__enumext_item_number_int - \g__enumext_item_anskey_int }
2223 }
2224 }

```

(End of definition for \\_\_enumext\_item\_answer\_diff:.)

\\_\_enumext\_check\_ans\_show:

The function \\_\_enumext\_check\_ans\_show: will be executed within the function \\_\_enumext\_execute\_after\_env: when the key `check-ans` is active, that is, when \g\_\_enumext\_check\_ans\_key\_bool is “*true*” and will return the appropriate message according to the value of \g\_\_enumext\_item\_answer\_diff\_int set by the function \\_\_enumext\_item\_answer\_diff:.

```

2225 \cs_new_protected:Nn __enumext_check_ans_show:
2226 {
2227 \int_case:nn { \g__enumext_item_answer_diff_int }
2228 {
2229 { -1 } { __enumext_check_ans_msg_less: }
2230 { 0 } { __enumext_check_ans_msg_same_ok: }
2231 { 1 } { __enumext_check_ans_msg_greater: }
2232 }
2233 }
2234 \cs_new_protected:Nn __enumext_check_ans_msg_less:
2235 {
2236 \msg_warning:nneee { enumext } { item-less-answer } { \g__enumext_store_name_tl }
2237 { \g__enumext_envir_name_tl } { \g__enumext_start_line_tl }
2238 }
2239 \cs_new_protected:Nn __enumext_check_ans_msg_same_ok:
2240 {
2241 \msg_term:nneee { enumext } { items-same-answer } { \g__enumext_store_name_tl }
2242 { \g__enumext_envir_name_tl } { \g__enumext_start_line_tl }
2243 }
2244 \cs_new_protected:Nn __enumext_check_ans_msg_greater:
2245 {
2246 \msg_warning:nneee { enumext } { item-greater-answer } { \g__enumext_store_name_tl }
2247 { \g__enumext_envir_name_tl } { \g__enumext_start_line_tl }
2248 }

```

(End of definition for \\_\_enumext\_check\_ans\_show: and others.)

\\_\_enumext\_check\_ans\_log:

The function \\_\_enumext\_check\_ans\_log: will be executed within the function \\_\_enumext\_execute\_after\_env: when the key `check-ans` is not active, that is, when \g\_\_enumext\_check\_ans\_key\_bool is “*false*” and write in the log the appropriate message according to the value of \g\_\_enumext\_item\_answer\_diff\_int set by the function \\_\_enumext\_item\_answer\_diff:.

```

2249 \cs_new_protected:Nn __enumext_check_ans_log:
2250 {
2251 \int_case:nn { \g__enumext_item_answer_diff_int }
2252 {
2253 { -1 } { __enumext_check_ans_log_msg_less: }
2254 { 0 } { __enumext_check_ans_log_msg_same_ok: }
2255 { 1 } { __enumext_check_ans_log_msg_greater: }
2256 }
2257 }
2258 \cs_new_protected:Nn __enumext_check_ans_log_msg_less:
2259 {
2260 \msg_log:nneee { enumext } { item-less-answer } { \g__enumext_store_name_tl }
2261 { \g__enumext_envir_name_tl } { \g__enumext_start_line_tl }
2262 }
2263 \cs_new_protected:Nn __enumext_check_ans_log_msg_same_ok:
2264 {
2265 \msg_log:nneee { enumext } { items-same-answer } { \g__enumext_store_name_tl }
2266 { \g__enumext_envir_name_tl } { \g__enumext_start_line_tl }
2267 }
2268 \cs_new_protected:Nn __enumext_check_ans_log_msg_greater:
2269 {
2270 \msg_log:nneee { enumext } { item-greater-answer } { \g__enumext_store_name_tl }
2271 { \g__enumext_envir_name_tl } { \g__enumext_start_line_tl }
2272 }

```

(End of definition for \\_\_enumext\_check\_ans\_log: and others.)

### 13.26.6 Check for \item\* and \anspic\* commands

\\_\_enumext\_check\_starred\_cmd:n

The function \\_\_enumext\_check\_starred\_cmd:n performs an *extra check* for the `keyans`, `keyans*` and `keyanspic` environments. Unlike the *check* executed by `check-ans` key this one is not controlled by any key, it is intended to prevent the forgetting of `\item*` or `\anspic*` in these environments.

```

2273 \cs_new_protected:Npn __enumext_check_starred_cmd:n #1
2274 {
2275 \int_compare:nNnT
2276 { \g__enumext_check_starred_cmd_int } = { 0 }
2277 {
2278 \msg_warning:nnnV
2279 { enumext } { missing-starred } { #1 } \l__enumext_check_start_line_env_tl
2280 }
2281 \int_compare:nNnT
2282 { \g__enumext_check_starred_cmd_int } > { 1 }
2283 {
2284 \msg_warning:nnnV
2285 { enumext } { many-starred } { #1 } \l__enumext_check_start_line_env_tl
2286 }
2287 \int_gzero:N \g__enumext_check_starred_cmd_int
2288 \tl_clear:N \l__enumext_check_start_line_env_tl
2289 }

```

(End of definition for \\_\_enumext\_check\_starred\_cmd:n.)

## 13.27 Keys and functions associated with storage

We add the keys `wrap-ans`, `wrap-opt`, `wrap-key`, `save-sep`, `mark-ans`, `mark-pos`, `show-ans`, `show-pos`, `mark-ref` and `save-ref` related to the “*storage system*” and internal mechanism of “*label and ref*” only at the *first level* of `enumext` and `enumext*`.

```

2290 \cs_set_protected:Npn __enumext_tmp:n #1
2291 {
2292 \keys_define:nn { enumext / #1 }
2293 {
2294 wrap-ans .cs_set_protected:Np = __enumext_anskey_wrapper:n ##1,
2295 wrap-ans .initial:n =
2296 {
2297 \fbox{\parbox[t]{\dimeval{\itemwidth -2\fboxsep -2\fboxrule}}{##1}}
2298 },
2299 wrap-ans .value_required:n = true,
2300 wrap-opt .cs_set_protected:Np = __enumext_keyans_wrapper_opt:n ##1,
2301 wrap-opt .initial:n = [{##1}],
2302 wrap-opt .value_required:n = true,
2303 wrap-key .cs_set_protected:Np = __enumext_keyans_wrapper_item:n ##1,
2304 wrap-key .value_required:n = true,

```

```

2305 save-sep .tl_set:N = \l__enumext_store_keyans_item_opt_sep_tl,
2306 save-sep .initial:n = {, ~ },
2307 save-sep .value_required:n = true,
2308 mark-ans .tl_set:N = \l__enumext_mark_answer_sym_tl,
2309 mark-ans .initial:n = \textasteriskcentered,
2310 mark-ans .value_required:n = true,
2311 mark-pos .choice:,
2312 mark-pos / left .code:n = \str_set:Nn \l__enumext_mark_position_str { l },
2313 mark-pos / right .code:n = \str_set:Nn \l__enumext_mark_position_str { r },
2314 mark-pos / unknown .code:n =
2315 \msg_error:nnee { enumext } { unknown-choice }
2316 { mark-pos } { left, ~ right } { \exp_not:n {##1} },
2317 mark-pos .initial:n = right,
2318 mark-pos .value_required:n = true,
2319 show-ans .bool_set:N = \l__enumext_show_answer_bool,
2320 show-ans .initial:n = false,
2321 show-ans .value_required:n = true,
2322 show-pos .bool_set:N = \l__enumext_show_position_bool,
2323 show-pos .initial:n = false,
2324 show-pos .value_required:n = true,
2325 mark-ref .tl_set:N = \l__enumext_mark_ref_sym_tl,
2326 mark-ref .initial:n = \textreferencemark,
2327 mark-ref .value_required:n = true,
2328 save-ref .bool_set:N = \l__enumext_store_ref_key_bool,
2329 save-ref .initial:n = false,
2330 save-ref .value_required:n = true,
2331 }
2332 }
2333 \clist_map_inline:nn { level-1, enumext* } { __enumext_tmp:n {##1} }

```

(End of definition for wrap-ans and others.)

For the `keyans` and `keyans*` environments we will only add the keys `save-sep`, `wrap-opt`, `wrap-key`, `mark-pos`, `show-ans` and `show-pos`.

```

2334 \cs_set_protected:Npn __enumext_tmp:n #1
2335 {
2336 \keys_define:nn { enumext / #1 }
2337 {
2338 save-sep .tl_set:N = \l__enumext_store_keyans_item_opt_sep_tl,
2339 save-sep .value_required:n = true,
2340 wrap-opt .cs_set_protected:Np = __enumext_keyans_wrapper_opt:n ##1,
2341 wrap-opt .value_required:n = true,
2342 wrap-key .cs_set_protected:Np = __enumext_keyans_wrapper_item:n ##1,
2343 wrap-key .value_required:n = true,
2344 mark-pos .choice:,
2345 mark-pos / left .code:n = \str_set:Nn \l__enumext_mark_position_str { l },
2346 mark-pos / right .code:n = \str_set:Nn \l__enumext_mark_position_str { r },
2347 mark-pos .initial:n = right,
2348 mark-pos .value_required:n = true,
2349 show-ans .bool_set:N = \l__enumext_show_answer_bool,
2350 show-ans .initial:n = false,
2351 show-ans .value_required:n = true,
2352 show-pos .bool_set:N = \l__enumext_show_position_bool,
2353 show-pos .initial:n = false,
2354 show-pos .value_required:n = true,
2355 }
2356 }
2357 \clist_map_inline:nn { keyans, keyans* } { __enumext_tmp:n {##1} }

```

(End of definition for save-sep and others.)

### 13.27.1 Storing structure of the environments

The idea behind “*storing structure*” in the *sequence* is to have a copy of the *structure of the environment* in which the key `save-ans` is being executed so we must capture the *optional argument* passed to the levels of the environment in which it is executed and “*storing*” this in the *sequence*.

```

__enumext_store_active_keys:n
__enumext_store_active_keys_vii:n

```

The functions `\__enumext_store_active_keys:n` and `\__enumext_store_active_keys_vii:n` will be responsible for the “*storing keys*” filtered from the *optional argument* of the environment in which the key `save-ans` is executed and the levels within this for the `enumext` and `enumext*` environments. We will execute this function only if the variable `\l__enumext_store_save_key_X_bool` is false, that is, the key

`store-key` is not active, establishing the variable `\__enumext_store_save_key_X_tl` with the filtered `<keys>`.

```

2358 \cs_new_protected:Npn __enumext_store_active_keys:n #1
2359 {
2360 \bool_if:cF { __enumext_store_save_key_ __enumext_level: _bool }
2361 {
2362 \tl_clear:c { __enumext_save_key_ __enumext_level: _tl }
2363 \tl_set:ce
2364 { __enumext_store_save_key_ __enumext_level: _tl }
2365 { __enumext_filter_save_key:n {#1} }
2366 }
2367 }
2368 \cs_new_protected:Npn __enumext_store_active_keys_vii:n #1
2369 {
2370 \bool_if:NF __enumext_store_save_key_vii_bool
2371 {
2372 \tl_clear:N __enumext_store_save_key_vii_tl
2373 \tl_set:Ne __enumext_store_save_key_vii_tl { __enumext_filter_save_key:n {#1} }
2374 }
2375 }

```

(End of definition for `\__enumext_store_active_keys:n` and `\__enumext_store_active_keys_vii:n`.)

### 13.27.2 Setting save-key key

Since this “*storing structure*” in the *sequence* established by the `save-ans` key when executing `\anskey` or `anskey*`, we will not be able to modify it. The best thing here is to have a key that allows you to modify the *optional argument* of the “*storing structure*” in the *sequence*.

`save-key` The values set by this key passed in the *optional argument* of the `enumext` and `enumext*` environments will override the values of the `\__enumext_store_save_key_X_tl` variable set by the functions `\__enumext_store_active_keys:n` and `\__enumext_store_active_keys_vii:n`. Now define the key `save-key` for all levels of `enumext` and `enumext*` environments.

```

2376 \cs_set_protected:Npn __enumext_tmp:n #1
2377 {
2378 \keys_define:nn { enumext / enumext* }
2379 {
2380 save-key .code:n = __enumext_parse_save_key_vii:n {##1},
2381 save-key .value_required:n = true,
2382 }
2383 \keys_define:nn { enumext / #1 }
2384 {
2385 save-key .code:n = __enumext_parse_save_key:n {##1},
2386 save-key .value_required:n = true,
2387 }
2388 }
2389 \clist_map_inline:nn { level-1, level-2, level-3, level-4 } { __enumext_tmp:n {#1} }

```

(End of definition for `save-key`.)

`\__enumext_parse_save_key:n` The functions `\__enumext_parse_save_key:n` and `\__enumext_parse_save_key_vii:n` will be responsible for “*storing keys*” in the variable `\__enumext_store_save_key_X_tl` for `enumext` and `enumext*`.

```

2390 \cs_new_protected:Npn __enumext_parse_save_key:n #1
2391 {
2392 \bool_set_true:c { __enumext_store_save_key_ __enumext_level: _bool }
2393 \tl_clear:c { __enumext_save_key_ __enumext_level: _tl }
2394 \tl_set:ce
2395 { __enumext_store_save_key_ __enumext_level: _tl }
2396 { __enumext_filter_save_key:n {#1} }
2397 }
2398 \cs_new_protected:Npn __enumext_parse_save_key_vii:n #1
2399 {
2400 \bool_set_true:N __enumext_store_save_key_vii_bool
2401 \tl_clear:N __enumext_store_save_key_vii_tl
2402 \tl_set:Ne __enumext_store_save_key_vii_tl { __enumext_filter_save_key:n {#1} }
2403 }

```

(End of definition for `\__enumext_parse_save_key:n` and `\__enumext_parse_save_key_vii:n`.)

### 13.27.3 Internal functions to store optional arguments

The function `\__enumext_filter_save_key:n` will be in charge of “*filtering keys*” we want to *stored* in *sequence* where `{#1}` represents the *optional argument* passed to the environment.

```

2404 \cs_new:Npn __enumext_filter_save_key:n #1
2405 {
2406 \use:e
2407 {
2408 \keyval_parse:NNn
2409 __enumext_filter_save_key_key:n
2410 __enumext_filter_save_key_pair:nn {#1}
2411 }
2412 }

```

The function `\__enumext_filter_save_key_key:n` will be responsible for “*filtering keys*” that are passed “*without value*” by excluding the `resume`, `resume*`, `no-store` and `base-fix` keys.

```

2413 \cs_new:Npn __enumext_filter_save_key_key:n #1
2414 {
2415 \str_case:nnF {#1}
2416 {
2417 { resume } {} { resume* } {} { no-store } {} { base-fix } {}
2418 }
2419 { , { \exp_not:n {#1} } }
2420 }

```

The function `\__enumext_filter_save_key_pair:nn` will be responsible for “*filtering keys*” that are passed “*with value*” by excluding the `series`, `resume`, `save-ans`, `save-ref`, `check-ans`, `show-ans`, `save-pos`, `wrap-ans`, `mark-ans`, `wrap-opt`, `save-sep`, `mark-ref`, `mini-env`, `mini-sep`, `mini-right` and `mini-right*` keys.

```

2421 \cs_new:Npn __enumext_filter_save_key_pair:nn #1#2
2422 {
2423 \str_case:nnF {#1}
2424 {
2425 { series } {} { resume } {} { save-ans } {} { save-ref } {}
2426 { save-key } {} { check-ans } {} { show-ans } {} { show-pos } {}
2427 { wrap-ans } {} { mark-ans } {} { wrap-opt } {} { save-sep } {}
2428 { mark-ref } {} { mini-env } {} { mini-sep } {} { mini-right } {}
2429 { mini-right* } {}
2430 }
2431 { , { \exp_not:n {#1} } } = { \exp_not:n {#2} } }
2432 }

```

(End of definition for `\__enumext_filter_save_key:n`, `\__enumext_filter_save_key_key:n`, and `\__enumext_filter_save_key_pair:nn`.)

### 13.27.4 Function for storing content in prop list

The function `\__enumext_store_addto_prop:n` stores the `{\content}` in *prop list* defined by `save-ans` key. The “*stored content*” is retrieved by means of the `\getkeyans` command.

The form in which the `{\content}` is “*stored*” in the *prop list* is `{\position}{\content}`. This function is used by `\anskey` in `enumext` and `enumext*` environments, `\item*` in `keyans` and `keyans*` environments and `\anspic*` in `keyanspic` environment.

```

2433 \cs_new_protected:Npn __enumext_store_addto_prop:n #1
2434 {
2435 \prop_gput_if_not_in:cen { g__enumext_ \l__enumext_store_name_tl _prop }
2436 {
2437 \int_eval:n { \prop_count:c { g__enumext_ \l__enumext_store_name_tl _prop } + 1 }
2438 }
2439 { #1 }
2440 }
2441 \cs_generate_variant:Nn __enumext_store_addto_prop:n { V }

```

(End of definition for `\__enumext_store_addto_prop:n`.)

### 13.27.5 Function for storing content in sequence

The function `\__enumext_store_addto_seq:n` stores the `{\content}` in *sequence* defined by `save-ans` key. This function is used by `\anskey` in `enumext`, `\item*` in `keyans` and `\anspic` in `keyanspic`.

The form in which the `{\content}` is stored in *sequence* is in a internal `enumext` or `enumext*` environments with the “*same structure*” in which the command was executed.

The “*stored content*” is retrieved by means of the `\printkeyans` command.

```

2442 \cs_new_protected:Npn __enumext_store_addto_seq:n #1
2443 {

```

```

2444 \seq_gput_right:cn { g__enumext_ \l__enumext_store_name_tl _seq } { #1 }
2445 }
2446 \cs_generate_variant:Nn __enumext_store_addto_seq:n { v, V }

```

(End of definition for \\_\_enumext\_store\_addto\_seq:n.)

### 13.27.6 Functions for storing structure in the sequence

The “*storing structure*” is handled by the functions \\_\_enumext\_store\_level\_open: and \\_\_enumext\_store\_level\_close: which are executed per level within the enumext environment.

```

2447 \cs_new_protected:Nn __enumext_store_level_open:
2448 {
2449 \bool_if:NT \l__enumext_check_answers_bool
2450 {
2451 \tl_if_empty:cTF { l__enumext_store_save_key_ __enumext_level: _tl }
2452 {
2453 __enumext_store_addto_seq:n
2454 {
2455 \item \begin{enumext}
2456 }
2457 }
2458 {
2459 \tl_put_left:cn { l__enumext_store_save_key_ __enumext_level: _tl }
2460 {
2461 \item \begin{enumext} [
2462 }
2463 \tl_put_right:cn { l__enumext_store_save_key_ __enumext_level: _tl }
2464 {
2465]
2466 }
2467 __enumext_store_addto_seq:v { l__enumext_store_save_key_ __enumext_level: _tl }
2468 }
2469 }
2470 }
2471 \cs_new_protected:Nn __enumext_store_level_close:
2472 {
2473 \bool_if:NT \l__enumext_check_answers_bool
2474 {
2475 __enumext_store_addto_seq:n { \end{enumext} }
2476 }
2477 }

```

(End of definition for \\_\_enumext\_store\_level\_open: and \\_\_enumext\_store\_level\_close:.)

The “*storing structure*” is handled by the functions \\_\_enumext\_store\_level\_open\_vii: and \\_\_enumext\_store\_level\_close\_vii: which are executed in the enumext\* environment.

```

2478 \cs_new_protected:Nn __enumext_store_level_open_vii:
2479 {
2480 \bool_if:NT \l__enumext_check_answers_bool
2481 {
2482 \tl_if_empty:NTF \l__enumext_store_save_key_vii_tl
2483 {
2484 __enumext_store_addto_seq:n
2485 {
2486 \item \begin{enumext*}
2487 }
2488 }
2489 {
2490 \tl_put_left:Nn \l__enumext_store_save_key_vii_tl
2491 {
2492 \item \begin{enumext*}[
2493 }
2494 \tl_put_right:Nn \l__enumext_store_save_key_vii_tl
2495 {
2496]
2497 }
2498 __enumext_store_addto_seq:V \l__enumext_store_save_key_vii_tl
2499 }
2500 }
2501 }
2502 \cs_new_protected:Nn __enumext_store_level_close_vii:
2503 {

```

```

2504 \bool_if:NT \l__enumext_check_answers_bool
2505 {
2506 __enumext_store_addto_seq:n { \end{enumext*} }
2507 }
2508 }

```

(End of definition for \\_\_enumext\_store\_level\_open\_vii: and \\_\_enumext\_store\_level\_close\_vii:.)

### 13.27.7 Function for show marks and position

```

__enumext_print_keyans_box:NN
__enumext_print_keyans_box:cc

```

The function \\_\_enumext\_print\_keyans\_box:NN print a box in the left margin with \l\_\_enumext\_mark\_answer\_sym\_tl used by the wrap-ans, show-ans and show-pos keys. The function takes two arguments:

#1: \l\_\_enumext\_labelwidth\_X\_dim  
 #2: \l\_\_enumext\_labelsep\_X\_dim

```

2509 \cs_new_protected:Nn __enumext_print_keyans_box:NN
2510 {
2511 \mode_leave_vertical:
2512 \skip_horizontal:n { -\dim_use:N #2 }
2513 \makebox[0pt][r]
2514 {
2515 \makebox[\dim_use:N #1][\l__enumext_mark_position_str]
2516 {
2517 \tl_use:N \l__enumext_mark_answer_sym_tl
2518 }
2519 }
2520 \skip_horizontal:n { \dim_use:N #2 }
2521 }
2522 \cs_generate_variant:Nn __enumext_print_keyans_box:NN { cc }

```

(End of definition for \\_\_enumext\_print\_keyans\_box:NN.)

### 13.28 The internal label and ref

The function \\_\_enumext\_store\_internal\_ref: handles the “internal label and ref” system used by the save-ref and mark-ref keys for \anskey will allow to execute \ref{⟨store name : position⟩} and will return 1.(a).i.A.

```

__enumext_store_internal_ref:

```

First we will remove the dots “.” from the current ⟨labels⟩, we do not want to get double dots in our references, then we will place this in the variable \l\_\_enumext\_newlabel\_arg\_two\_tl.

```

2523 \cs_new_protected:Nn __enumext_store_internal_ref:
2524 {
2525 \cs_set_protected:Npn __enumext_tmp:n ##1
2526 {
2527 \tl_set_eq:cc { \l__enumext_label_copy_##1_tl } { \l__enumext_label_##1_tl }
2528 \tl_reverse:c { \l__enumext_label_copy_##1_tl }
2529 \tl_remove_once:cn { \l__enumext_label_copy_##1_tl } { . }
2530 \tl_reverse:c { \l__enumext_label_copy_##1_tl }
2531 }
2532 \clist_map_inline:nn { i, ii, iii, iv, vii } { __enumext_tmp:n {##1} }
2533 \cs_set:Npn __enumext_tmp:n ##1
2534 { . \tl_use:c { \l__enumext_label_copy_ \int_to_roman:n {##1} _tl } }

```

Here we need to analyse the cases where the environment is started with enumext\* and if \anskey or anskey\* is running alone in it or if it is running in a nested enumext environment within the starting environment.

```

2535 \bool_lazy_all:nT
2536 {
2537 { \bool_if_p:N \g__enumext_starred_bool }
2538 { \int_compare_p:nNn { \l__enumext_level_int } = { 0 } }
2539 }
2540 {
2541 \tl_put_right:Ne \l__enumext_newlabel_arg_two_tl
2542 { \tl_use:N \l__enumext_label_copy_vii_tl }
2543 }
2544 \bool_lazy_all:nT
2545 {
2546 { \bool_not_p:n { \g__enumext_standar_bool } }
2547 { \bool_if_p:N \l__enumext_standar_bool }
2548 { \int_compare_p:nNn { \l__enumext_level_int } > { 0 } }
2549 }
2550 {
2551 \tl_put_right:Ne \l__enumext_newlabel_arg_two_tl
2552 {

```



```

2553 \tl_use:N \l__enumext_label_copy_vii_tl
2554 \int_step_function:nnN { 1 } { \l__enumext_level_int } \l__enumext_tmp:n
2555 }
2556 }

```

If started with `enumext` and if `\anskey` or `anskey*` is running alone in it or if it is running in a nested `enumext*` environment within the starting environment.

```

2557 \bool_lazy_all:nT
2558 {
2559 { \bool_if_p:N \g__enumext_standar_bool }
2560 { \int_compare_p:nNn { \l__enumext_level_int } > { 0 } }
2561 { \int_compare_p:nNn { \l__enumext_level_h_int } = { 0 } }
2562 }
2563 {
2564 \tl_put_right:Ne \l__enumext_newlabel_arg_two_tl
2565 {
2566 \tl_use:N \l__enumext_label_copy_i_tl
2567 \int_step_function:nnN { 2 } { \l__enumext_level_int } \l__enumext_tmp:n
2568 }
2569 }
2570 \cs_set:Npn \l__enumext_tmp:n ##1
2571 { \tl_use:c { l__enumext_label_copy_ \int_to_roman:n {##1} _tl } . }
2572 \bool_lazy_all:nT
2573 {
2574 { \bool_if_p:N \g__enumext_standar_bool }
2575 { \bool_if_p:N \l__enumext_starred_bool }
2576 { \int_compare_p:nNn { \l__enumext_level_int } > { 0 } }
2577 }
2578 {
2579 \tl_put_right:Ne \l__enumext_newlabel_arg_two_tl
2580 {
2581 \int_step_function:nnN { 1 } { \l__enumext_level_int } \l__enumext_tmp:n
2582 \tl_use:N \l__enumext_label_copy_vii_tl
2583 }
2584 }

```

Now we set the variable `\l__enumext_newlabel_arg_one_tl` which will contain `{(store name : position)}`.

```

2585 \tl_put_right:Ne \l__enumext_newlabel_arg_one_tl
2586 {
2587 \l__enumext_store_name_tl \c_colon_str
2588 \int_eval:n { \prop_count:c { g__enumext_ \l__enumext_store_name_tl _prop } }
2589 }

```

Now execute the function `\l__enumext_newlabel:nn` and save the result in the variable `\l__enumext_write_aux_file_tl` and finally we write in the `.aux` file.

```

2590 \tl_put_right:Ne \l__enumext_write_aux_file_tl
2591 {
2592 \l__enumext_newlabel:nn
2593 { \exp_not:V \l__enumext_newlabel_arg_one_tl }
2594 { \l__enumext_newlabel_arg_two_tl }
2595 }
2596 \l__enumext_write_aux_file_tl
2597 }

```

(End of definition for `\l__enumext_store_internal_ref:`)

### 13.29 Common functions for `\anskey` and `anskey*` environment

`\l__enumext_store_anskey_code:n`

The internal function `\l__enumext_store_anskey_code:n` first we pass the `{(argument)}` to the *prop list*, then checks the state of the variable `\l__enumext_store_ref_key_bool` handled by the *save-ref* key and will call the function `\l__enumext_store_internal_ref:` for the “*internal label and ref*” system. Followed by this if the *show-ans* or *show-pos* keys are active we will show the “*wrapped*” `{(argument)}`.

```

2598 \cs_new_protected:Npn \l__enumext_store_anskey_code:n #1
2599 {
2600 \int_gincr:N \g__enumext_item_anskey_int
2601 \l__enumext_store_addto_prop:n {#1}
2602 \bool_if:NT \l__enumext_store_ref_key_bool
2603 {
2604 \l__enumext_store_internal_ref:
2605 }
2606 \l__enumext_anskey_show_wrap_left:n { #1 }

```

Now we start processing the  $[key = val]$  passed to the command to build our `\item` in the variable `\l__enumext_store_anskey_arg_tl` which we will “store” in the *sequence*. First we clear the variable `\l__enumext_store_anskey_arg_tl` and process the  $\langle keys \rangle$ , if the `break-col` key is present and the command is running under `enumext` (not in `enumext*`) we will add `\columnbreak` and then `\item`.

```

2607 \tl_clear:N \l__enumext_store_anskey_arg_tl
2608 \bool_lazy_and:nnT
2609 { \bool_if_p:N \l__enumext_store_columns_break_bool }
2610 { \bool_not_p:n { \l__enumext_starred_bool } }
2611 {
2612 \tl_put_left:Nn \l__enumext_store_anskey_arg_tl { \columnbreak }
2613 }
2614 \tl_put_right:Nn \l__enumext_store_anskey_arg_tl { \item }

```

If the `item-join` key is present and the command is running under `enumext*` we will add  $(\langle number \rangle)$  to `\l__enumext_store_anskey_arg_tl`.

```

2615 \bool_lazy_and:nnT
2616 { \bool_not_p:n { \l__enumext_starred_bool } }
2617 { \int_compare_p:nNn { \l__enumext_store_item_join_int } > { 1 } }
2618 {
2619 \tl_put_right:Ne \l__enumext_store_anskey_arg_tl
2620 {
2621 (\exp_not:V \l__enumext_store_item_join_int)
2622 }
2623 }

```

And now we will review the keys `item-star`, `item-sym*` and `item-pos*` and pass them to `\l__enumext_store_anskey_arg_tl` along with the  $\{\langle argument \rangle\}$  for `\anskey` or  $\langle body \rangle$  for `anskey*`.

```

2624 \bool_if:NTF \l__enumext_store_item_star_bool
2625 {
2626 \tl_put_right:Nn \l__enumext_store_anskey_arg_tl { * }
2627 \tl_if_empty:NF \l__enumext_store_item_symbol_tl
2628 {
2629 \tl_put_right:Ne \l__enumext_store_anskey_arg_tl
2630 {
2631 [\exp_not:V \l__enumext_store_item_symbol_tl]
2632 }
2633 }
2634 \dim_compare:nT
2635 {
2636 \l__enumext_store_item_symbol_sep_dim != \c_zero_dim
2637 }
2638 {
2639 \tl_put_right:Ne \l__enumext_store_anskey_arg_tl
2640 {
2641 [\exp_not:V \l__enumext_store_item_symbol_sep_dim]
2642 }
2643 }
2644 \tl_put_right:Nn \l__enumext_store_anskey_arg_tl {#1}
2645 }
2646 {
2647 \tl_put_right:Nn \l__enumext_store_anskey_arg_tl {#1}
2648 }

```

Finally we check if the `save-ref` key are active along with the `hyperref` package load, if both conditions are met, it will create the `\hyperlink` with “symbol” set by `mark-ref` key and then store in *sequence*.

```

2649 \bool_lazy_and:nnT
2650 { \bool_if_p:N \l__enumext_store_ref_key_bool }
2651 { \bool_if_p:N \l__enumext_hyperref_bool }
2652 {
2653 \tl_put_right:Ne \l__enumext_store_anskey_arg_tl
2654 {
2655 \hfill \exp_not:N \hyperlink { \exp_not:V \l__enumext_newlabel_arg_one_tl }
2656 { \exp_not:V \l__enumext_mark_ref_sym_tl }
2657 }
2658 }
2659 \l__enumext_store_addto_seq:V \l__enumext_store_anskey_arg_tl
2660 }

```

(End of definition for `\l__enumext_store_anskey_code:n`.)

`\l__enumext_anskey_show_wrap_arg:n`

The function `\l__enumext_anskey_show_wrap_arg:n` “wraps” the  $\{\langle argument \rangle\}$  passed to `\anskey` and the  $\langle body \rangle$  for `anskey*` when using the `wrap-ans` key.

```

2661 \cs_new_protected:Npn __enumext_anskey_show_wrap_arg:n #1
2662 {
2663 \par
2664 \bool_if:NTF \l__enumext_starred_bool
2665 {
2666 __enumext_print_keyans_box:NN
2667 \l__enumext_labelwidth_vii_dim \l__enumext_labelsep_vii_dim
2668 }
2669 {
2670 __enumext_print_keyans_box:cc
2671 { \l__enumext_labelwidth_ __enumext_level: _dim }
2672 { \l__enumext_labelsep_ __enumext_level: _dim }
2673 }
2674 __enumext_anskey_wrapper:n { #1 }
2675 }

```

(End of definition for \\_\_enumext\_anskey\_show\_wrap\_arg:n.)

\\_\_enumext\_anskey\_show\_wrap\_left:n

The function \\_\_enumext\_anskey\_show\_wrap\_left:n will show the “*mark*” defined by the `mark-ans` key or the “*position*” of the  $\langle content \rangle$  stored in the *prop list* when using the `show-pos` key on the left margin next to the “*wraps*”  $\langle argument \rangle$  passed to `\anskey` and the  $\langle body \rangle$  in `anskey*` on the right side when using the `show-ans` key.

```

2676 \cs_new_protected:Npn __enumext_anskey_show_wrap_left:n #1
2677 {
2678 \bool_if:NT \l__enumext_show_answer_bool
2679 {
2680 __enumext_anskey_show_wrap_arg:n { #1 }
2681 }
2682 \bool_if:NT \l__enumext_show_position_bool
2683 {
2684 \tl_set:Nx \l__enumext_mark_answer_sym_tl
2685 {
2686 \group_begin:
2687 \exp_not:N \normalfont
2688 \exp_not:N \footnotesize [\int_eval:n
2689 {
2690 \prop_count:c { g__enumext_ \l__enumext_store_name_tl _prop }
2691 }
2692]
2693 \group_end:
2694 }
2695 __enumext_anskey_show_wrap_arg:n { #1 }
2696 }
2697 }

```

(End of definition for \\_\_enumext\_anskey\_show\_wrap\_left:n.)

### 13.30 The command \anskey

Since we will be “*storing content*” in a `list` environment within *sequences* and can (more or less) manage the options passed to each level, it is necessary that we have a little more control over `\item` when storing.

The `\anskey` command will cover this point and give it similar behaviour to that of `\item` in the `enumext` and `enumext*` environments executed as follows `\anskey[⟨key = val⟩]{⟨content⟩}`.

First we’ll add the keys `break-col`, `item-join`, `item-star`, `item-sym*` and `item-pos*`.

```

2698 \keys_define:nn { enumext / anskey }
2699 {
2700 break-col .bool_set:N = \l__enumext_store_columns_break_bool,
2701 break-col .default:n = true,
2702 break-col .value_forbidden:n = true,
2703 item-join .int_set:N = \l__enumext_store_item_join_int,
2704 item-join .value_required:n = true,
2705 item-star .bool_set:N = \l__enumext_store_item_star_bool,
2706 item-star .default:n = true,
2707 item-star .value_forbidden:n = true,
2708 item-sym* .tl_set:N = \l__enumext_store_item_symbol_tl,
2709 item-sym* .value_required:n = true,
2710 item-pos* .dim_set:N = \l__enumext_store_item_symbol_sep_dim,
2711 item-pos* .value_required:n = true,
2712 unknown .code:n = { __enumext_anskey_unknown:n {#1} },
2713 }

```

The `\keys` are stored in `\l_keys_key_str` and the value (if any) is passed as an argument to the function `\__enumext_anskey_unknown:n`.

```

2714 \cs_new_protected:Npn __enumext_anskey_unknown:n #1
2715 {
2716 \exp_args:NV __enumext_anskey_unknown:nn \l_keys_key_str {#1}
2717 }
2718 \cs_new_protected:Npn __enumext_anskey_unknown:nn #1 #2
2719 {
2720 \tl_if_blank:nTF {#2}
2721 {
2722 \msg_error:nnn { enumext } { anskey-cmd-key-unknown } {#1}
2723 }
2724 {
2725 \msg_error:nnnn { enumext } { anskey-cmd-key-value-unknown } {#1} {#2}
2726 }
2727 }

```

(End of definition for `\__enumext_anskey_unknown:n` and `\__enumext_anskey_unknown:nn`.)

- The `\anskey` command will only be present when using the `save-ans` key in `enumext` and `enumext*` environments, otherwise it will return an error.

`\anskey` We will first call the function `\__enumext_anskey_safe_outer:` to be sure where we execute the command, then we will check the state of the variable `\l__enumext_check_answers_bool` set by the key `no-store`, if is true we will increment `\g__enumext_item_anskey_int` for the internal “*check answer*” system and execute the function `\__enumext_anskey_safe_inner:n` to ensure that the command is not nested and that the argument is not empty, finally search the `[⟨key = val⟩]` and call the function `\__enumext_store_anskey_code:n`.

```

2728 \NewDocumentCommand \anskey { o +m }
2729 {
2730 __enumext_anskey_safe_outer:
2731 \group_begin:
2732 \bool_if:NT \l__enumext_check_answers_bool
2733 {
2734 \tl_if_novalue:nF {#1}
2735 {
2736 \keys_set:nn { enumext / anskey } {#1}
2737 }
2738 \tl_if_blank:nTF {#2}
2739 {
2740 \msg_error:nn { enumext } { anskey-empty-arg }
2741 }
2742 {
2743 __enumext_anskey_safe_inner:
2744 __enumext_store_anskey_code:n {#2}
2745 }
2746 }
2747 \group_end:
2748 }

```

(End of definition for `\anskey`. This function is documented on page 13.)

### 13.30.1 Internal functions for the command

`\__enumext_anskey_safe_outer:` The `\__enumext_store_anskey_safe_outer:` function will return the appropriate messages when the command is executed outside the environment in which the `save-ans` key was activated.

`\__enumext_anskey_safe_inner:`

```

2749 \cs_new_protected:Nn __enumext_anskey_safe_outer:
2750 {
2751 \bool_if:NF \l__enumext_store_active_bool
2752 {
2753 \msg_error:nnnn { enumext } { anskey-wrong-place } { anskey } { enumext }
2754 }
2755 \int_compare:nNt { \l__enumext_keyans_level_int } = { 1 }
2756 {
2757 \msg_error:nnnn { enumext } { command-wrong-place } { anskey } { keyans }
2758 }
2759 \int_compare:nNt { \l__enumext_keyans_level_h_int } = { 1 }
2760 {
2761 \msg_error:nnnn { enumext } { command-wrong-place } { anskey } { keyans* }
2762 }
2763 \int_compare:nNt { \l__enumext_keyans_pic_level_int } = { 1 }
2764 {

```

```

2765 \msg_error:nnnn { enumext } { command-wrong-place } { anskey } { keyanspic }
2766 }
2767 }

The __enumext_anskey_safe_inner: function will first check if the command is nested, if preceded by a
not numbered \item or if it is in math mode returning the appropriate messages.

2768 \cs_new_protected:Nn __enumext_anskey_safe_inner:
2769 {
2770 \int_incr:N __enumext_anskey_level_int
2771 \int_compare:nNt { __enumext_anskey_level_int } > { 1 }
2772 {
2773 \msg_error:nn { enumext } { anskey-nested }
2774 }
2775 \bool_if:NF __enumext_item_number_bool
2776 {
2777 \msg_error:nn { enumext } { anskey-unnumber-item }
2778 }
2779 \mode_if_math:T
2780 {
2781 \msg_error:nne { enumext } { anskey-math-mode } { \c_backslash_str anskey }
2782 }
2783 }

```

(End of definition for \\_\_enumext\_anskey\_safe\_outer: and \\_\_enumext\_anskey\_safe\_inner:.)

### 13.31 The environment anskey\*

Managing *verbatim content* in an environment is quite complicated, I learned that when creating the `scontents` package, so to be able to have support at this point it is best to play a little with the internal code of `scontents` and *hooks*. Some considerations I should have here before implementing this:

- If some package, class or user has defined the environment with the same name somewhere in the document it would be a problem, you would not know what argument has been passed to `store-env`, if you are using the key `print-env` or the `write-out` key, sure, I can detect and modify it within the `enumext` and `enumext*` environments, but it would look strange not to have some keys available when running within these environments.
- A better (perhaps a bit paranoid) option is to define it within the environment in which the `save-ans` key is executed. and have it available only when that key is executed, here I would have absolute control of the *(keys)* and I make sure that `write-out` is not used, then using *hooks after* I undefine it and using *hook before* I check if it has been created by any package, class or user and I return a error, then the user will have to see how to solve the problem.

\\_\_enumext\_undefine\_anskey\_env:

The function `\__enumext_undefine_anskey_env:` will undefine the environment `anskey*` and will be passed to the function `\__enumext_execute_after_env:` (§13.32) which is executed after the environment in which the key `save-ans` is active.

```

2784 \cs_new_protected:Nn __enumext_undefine_anskey_env:
2785 {
2786 \cs_undefine:c { anskey* }
2787 \cs_undefine:c { endanskey* }
2788 \cs_undefine:c { __scontents_anskey*_env_begin: }
2789 \cs_undefine:c { __scontents_anskey*_env_end: }
2790 }

```

Detection of the `anskey*` environment outside the `enumext` and `enumext*` environments.

```

2791 __enumext_before_env:nn { enumext }
2792 {
2793 \bool_lazy_and:nnT
2794 { \int_compare_p:nNn { __enumext_level_int } = { 0 } }
2795 { \int_compare_p:nNn { __enumext_level_h_int } = { 0 } }
2796 {
2797 \cs_if_free:cF { __scontents_anskey*_env_begin: }
2798 {
2799 \msg_error:nnn { enumext } { anskey-env-error } { anskey* }
2800 }
2801 }
2802 }
2803 __enumext_before_env:nn { enumext* }
2804 {
2805 \bool_lazy_and:nnT
2806 { \int_compare_p:nNn { __enumext_level_int } = { 0 } }
2807 { \int_compare_p:nNn { __enumext_level_h_int } = { 0 } }
2808 {

```

```

2809 \cs_if_free:cf { __scontents_anskey*_env_begin: }
2810 {
2811 \msg_error:nnn { enumext } { anskey-env-error } { anskey* }
2812 }
2813 }
2814 }

```

Detection of the `anskey*` environment inside the `keyans`, `keyans*` and `keyanspic` environments, if preceded by a not numbered `\item` or if it is in *math mode* returning the appropriate messages.

```

2815 __enumext_before_env:nn { anskey* }
2816 {
2817 \int_compare:nNt { __enumext_keyans_level_int } = { 1 }
2818 {
2819 \msg_error:nnn { enumext } { anskey-env-wrong } { keyans }
2820 }
2821 \int_compare:nNt { __enumext_keyans_level_h_int } = { 1 }
2822 {
2823 \msg_error:nnn { enumext } { anskey-env-wrong } { keyans* }
2824 }
2825 \int_compare:nNt { __enumext_keyans_pic_level_int } = { 1 }
2826 {
2827 \msg_error:nnn { enumext } { anskey-env-wrong } { keyanspic }
2828 }
2829 \bool_if:NF __enumext_item_number_bool
2830 {
2831 \msg_error:nn { enumext } { anskey-unnumber-item }
2832 }
2833 \mode_if_math:T
2834 {
2835 \msg_error:nnn { enumext } { anskey-math-mode } { anskey* }
2836 }
2837 }

```

(End of definition for `\__enumext_undefine_anskey_env:`)

`anskey*`

The function `\__enumext_anskey_env_make:n` creates the environment `anskey*` (custom version of `scontents` environment) by setting the initial keys `store-env={⟨store name⟩}` and `print-env=false`.

To maintain the *scope* of the environment and that it is only active when the key `save-ans` is active we will pass this function to the function `\__enumext_storing_exec:` (§13.26.1) and we will execute it only if the variable `\__enumext_anskey_env_bool` is true, with this we prevent it from being executed again when the environment is nested and the key `save-ans` is active, which returns an error for part of the package `scontents`.

```

2838 \cs_new_protected:Npn __enumext_anskey_env_make:n #1
2839 {
2840 \bool_if:NT __enumext_anskey_env_bool
2841 {
2842 \newenvsc{anskey*}[store-env=#1,print-env=false]
2843 __enumext_anskey_env_exec:
2844 }
2845 }
2846 \cs_generate_variant:Nn __enumext_anskey_env_make:n { V }

```

The function `\__enumext_anskey_env_define_keys:` will add the keys `break-col`, `item-join`, `item-join`, `item-star`, `item-sym*` and `item-pos*` and will leave the keys `print-env`, `store-env` and `write-out` undefined. We will apply this function using the *hook* function `\__enumext_before_env:nn`.

```

2847 \cs_new_protected:Nn __enumext_anskey_env_define_keys:
2848 {
2849 \keys_define:nn { scontents / scontents }
2850 {
2851 break-col .bool_gset:N = __enumext_store_columns_break_bool,
2852 break-col .default:n = true,
2853 break-col .value_forbidden:n = true,
2854 item-join .int_gset:N = __enumext_store_item_join_int,
2855 item-join .value_required:n = true,
2856 item-star .bool_gset:N = __enumext_store_item_star_bool,
2857 item-star .default:n = true,
2858 item-star .value_forbidden:n = true,
2859 item-sym* .tl_gset:N = __enumext_store_item_symbol_tl,
2860 item-sym* .value_required:n = true,
2861 item-pos* .dim_gset:N = __enumext_store_item_symbol_sep_dim,
2862 item-pos* .value_required:n = true,

```

```

2863 print-env .undefine:,
2864 store-env .undefine:,
2865 write-out .undefine:,
2866 unknown .code:n = { __enumext_anskey_env_unknown:n {##1} },
2867 }
2868 }

```

The *⟨keys⟩* are stored in `\l_keys_key_str` and the value (if any) is passed as an argument to the function `\__enumext_anskey_env_unknown:n`.

```

2869 \cs_new_protected:Npn __enumext_anskey_env_unknown:n #1
2870 {
2871 \exp_args:NV __enumext_anskey_env_unknown:nn \l_keys_key_str {#1}
2872 }
2873 \cs_new_protected:Npn __enumext_anskey_env_unknown:nn #1#2
2874 {
2875 \tl_if_blank:nTF {#2}
2876 {
2877 \msg_error:nnn { enumext } { anskey-env-key-unknown } {#1}
2878 }
2879 {
2880 \msg_error:nnnn { enumext } { anskey-env-key-value-unknown } {#1} {#2}
2881 }
2882 }

```

The function `\__enumext_anskey_env_reset_keys:` will leave the keys `break-col`, `item-join`, `item-join`, `item-star`, `item-sym*` and `item-pos*` undefined. We will apply this function using the *hook* function `\__enumext_after_env:nn`.

```

2883 \cs_new_protected:Nn __enumext_anskey_env_reset_keys:
2884 {
2885 \keys_define:nn { scontents / scontents }
2886 {
2887 break-col .undefine:,
2888 item-join .undefine:,
2889 item-star .undefine:,
2890 item-sym* .undefine:,
2891 item-pos* .undefine:,
2892 write-out .code:n = {
2893
2894 \bool_set_false:N \l__scontents_storing_bool
2895 \bool_set_true:N \l__scontents_writing_bool
2896 \tl_set:Nn \l__scontents_fname_out_tl {##1}
2897 },
2898 write-out .value_required:n = true,
2899 print-env .meta:nn = { scontents } { print-env = ##1 },
2900 print-env .default:n = true,
2901 store-env .meta:nn = { scontents } { store-env = ##1 },
2902 unknown .code:n = { __scontents_parse_environment_keys:n {##1} },
2903 }
2904 }

```

The function `\__enumext_rescan_anskey_env:n` will be responsible for bringing the *⟨body⟩* of the environment saved in the sequence `\g__scontents_name_⟨store name⟩_seq` to pass it to our *sequence* and *prop list*.

```

2904 \cs_new_protected:Npn __enumext_rescan_anskey_env:n #1
2905 {
2906 \group_begin:
2907 \int_set:Nn \tex_newlinechar:D { `^^J }
2908 __scontents_rescan_tokens:x
2909 {
2910 \endgroup % This assumes \catcode`\=0... Things might go off otherwise.
2911 #1
2912 }
2913 }

```

(End of definition for *anskey\** and others. This function is documented on page 13.)

`\__enumext_anskey_env_exec:`

The function `\__enumext_anskey_env_exec:` will be responsible for processing all the code necessary for the execution of the environment. The first thing will be to add our *⟨keys⟩*.

```

2914 \cs_new_protected:Nn __enumext_anskey_env_exec:
2915 {
2916 __enumext_before_env:nn { anskey* }
2917 {
2918 __enumext_anskey_env_define_keys:
2919 }

```



Now we will execute our actions after the `anskey*` environment is closed. We'll fetch the contents of the *environment body* that is now saved in `\g__scontents_name_⟨store name⟩_seq` and store it in the variable `\l__enumext_store_anskey_env_tl` then we execute the rest of the functions.

```

2920 \hook_if_empty:nF {env/anskey*/after}
2921 {
2922 \hook_gremove_code:nn {env/anskey*/after} { * }
2923 }
2924 __enumext_after_env:nn { anskey* }
2925 {
2926 __enumext_anskey_env_save_keys:
2927 \tl_clear:N \l__enumext_store_anskey_env_tl
2928 \tl_clear:N \l__enumext_store_anskey_opt_tl
2929 \bool_if:NT \l__enumext_check_answers_bool
2930 {
2931 \tl_gset:Ne \l__enumext_store_anskey_env_tl
2932 {
2933 \seq_item:ce { g__scontents_name_ \l__enumext_store_name_tl _seq } { -1 }
2934 }
2935 \regex_match:nVTF
2936 { ^\s* \z | ^\s* \u{c__scontents_hidden_space_str} \z }
2937 \l__enumext_store_anskey_env_tl
2938 {
2939 \msg_error:nn { enumext } { anskey-empty-arg }
2940 }
2941 {
2942 __enumext_anskey_env_store:
2943 }
2944 }
2945 __enumext_anskey_env_clean_vars:
2946 __enumext_anskey_env_reset_keys:
2947 }
2948 }

```

The use of `\hook_gremove_code:nn` is necessary here, otherwise the `{⟨code⟩}` passed to `\__enumext_after_env:nn{anskey*}` will be accumulated for each execution. The last function `\__enumext_anskey_env_reset_keys:` is necessary so as not to hinder any `scontents` environment running within `enumext` or `enumext*`.

(End of definition for `\__enumext_anskey_env_exec:`.)

```

__enumext_anskey_env_save_keys:
__enumext_anskey_env_store:
__enumext_anskey_env_clean_vars:

```

The function `\__enumext_anskey_env_save_keys:` processing the `[⟨key = val⟩]` passed to the environment and save this in the variable `\l__enumext_store_anskey_opt_tl`. If the `break-col` key is present and the environment is running under `enumext` (not in `enumext*`) we will add the key `break-col`.

```

2949 \cs_new_protected:Nn __enumext_anskey_env_save_keys:
2950 {
2951 \bool_lazy_and:nnT
2952 { \bool_if_p:N \g__enumext_store_columns_break_bool }
2953 { \bool_not_p:n { \l__enumext_starred_bool } }
2954 {
2955 \tl_put_left:Ne \l__enumext_store_anskey_opt_tl { ,break-col, }
2956 }

```

If the `item-join` key is present and the command is running under `enumext*` we will add to `\l__enumext_store_anskey_opt_tl`.

```

2957 \bool_lazy_and:nnT
2958 { \bool_not_p:n { \l__enumext_starred_bool } }
2959 { \int_compare_p:nNn { \g__enumext_store_item_join_int } > { 1 } }
2960 {
2961 \tl_put_left::Ne \l__enumext_store_anskey_opt_tl
2962 {
2963 ,item-join = \exp_not:V \g__enumext_store_item_join_int,
2964 }
2965 }

```

And now we will review the keys `item-star`, `item-sym*` and `item-pos*` and pass them to `\l__enumext_store_anskey_opt_tl`.

```

2966 \bool_if:NT \g__enumext_store_item_star_bool
2967 {
2968 \tl_put_left:Ne \l__enumext_store_anskey_opt_tl
2969 {
2970 ,item-star,
2971 }
2972 \tl_if_empty:NF \g__enumext_store_item_symbol_tl

```

```

2973 {
2974 \tl_put_left:Ne \l__enumext_store_anskey_opt_tl
2975 {
2976 ,item-sym* = \exp_not:V \g__enumext_store_item_symbol_tl,
2977 }
2978 }
2979 \dim_compare:nT
2980 {
2981 \g__enumext_store_item_symbol_sep_dim != \c_zero_dim
2982 }
2983 {
2984 \tl_put_left:Ne \l__enumext_store_anskey_opt_tl
2985 {
2986 ,item-pos* = \exp_not:V \g__enumext_store_item_symbol_sep_dim,
2987 }
2988 }
2989 }
2990 }

```

The function `\__enumext_anskey_env_store:` will be responsible for storing the content of the environment using the functions `\__enumext_store_anskey_code:n` and `\__enumext_rescan_anskey_env:n`.

```

2991 \cs_new_protected:Nn __enumext_anskey_env_store:
2992 {
2993 \group_begin:
2994 \tl_if_empty:NTF \l__enumext_store_anskey_opt_tl
2995 {
2996 \exp_args:Ne
2997 __enumext_store_anskey_code:n
2998 {
2999 __enumext_rescan_anskey_env:n { \l__enumext_store_anskey_env_tl }
3000 }
3001 }
3002 {
3003 \keys_set_known:nV { enumext / anskey } \l__enumext_store_anskey_opt_tl
3004 \exp_args:Ne
3005 __enumext_store_anskey_code:n
3006 {
3007 __enumext_rescan_anskey_env:n { \l__enumext_store_anskey_env_tl }
3008 }
3009 }
3010 \group_end:
3011 }

```

The function `\__enumext_anskey_env_clean_vars:` will return the global variables used by the `<keys>` to their initial state.

```

3012 \cs_new_protected:Nn __enumext_anskey_env_clean_vars:
3013 {
3014 \bool_gset_false:N \g__enumext_store_columns_break_bool
3015 \int_gzero:N \g__enumext_store_item_join_int
3016 \bool_gset_false:N \g__enumext_store_item_star_bool
3017 \tl_gclear:N \g__enumext_store_item_symbol_tl
3018 \dim_gzero:N \g__enumext_store_item_symbol_sep_dim
3019 }

```

(End of definition for `\__enumext_anskey_env_save_keys:`, `\__enumext_anskey_env_store:`, and `\__enumext_anskey_env_clean_vars:`.)

### 13.32 Executing anskey\*, check-ans and write .log

`\__enumext_execute_after_env:`

The `\__enumext_execute_after_env:` function will first return the appropriate message for the end of the environment in which the `save-ans` key is being executed, then call the `\__enumext_item_answer_diff:` function and then will write the values of the global variables used to the `.log` file. If the key `check-ans` is active it will execute the function `\__enumext_check_ans_show:` and show the result in the terminal, otherwise it will execute the function `\__enumext_check_ans_log:` and write the results in the `.log` file, undefine the environment `anskey*` (§13.31) through the function `\__enumext_undefine_anskey_env:` and finally we execute the function `\__enumext_reset_global_vars:` returning the used variables to their original state.

```

3020 \cs_new_protected:Nn __enumext_execute_after_env:
3021 {
3022 \int_compare:nNnT { \l__enumext_level_int } = { 0 }
3023 {
3024 \tl_if_empty:NF \g__enumext_store_name_tl

```

```

3025 {
3026 __enumext_stop_save_ans_msg:
3027 __enumext_item_answer_diff:
3028 __enumext_log_global_vars:
3029 __enumext_log_answer_vars:
3030 \bool_if:NTF __enumext_check_ans_key_bool
3031 {
3032 __enumext_check_ans_show:
3033 }
3034 { __enumext_check_ans_log: }
3035 __enumext_undefine_anskey_env:
3036 }
3037 __enumext_reset_global_vars:
3038 }
3039 }

```

(End of definition for \\_\_enumext\_execute\_after\_env:.)

- This function is passed to the function \\_\_enumext\_after\_env:nn for the environments `enumext` (§13.39) and `enumext*` (§13.44) and it is executed only when the environments are not nested or at some level of these..

### 13.33 Common functions for keyans, keyans\* and keyanspic

#### 13.33.1 Storing content in prop list

\\_\_enumext\_keyans\_addto\_prop:n

The function \\_\_enumext\_keyans\_addto\_prop:n will pass the the current *⟨label⟩* for *\item\** in *keyans* environment and the current *⟨label⟩* for *\anspic\** in *keyanspic* environment followed by the *⟨contents⟩* of the *optional argument* of both commands to the \\_\_enumext\_store\_current\_label\_tl variable, which will be stored to the *prop list* defined by the *save-ans* key using the function \\_\_enumext\_store\_addto\_prop:V.

```

3040 \cs_new_protected:Npn __enumext_keyans_addto_prop:n #1
3041 {
3042 \tl_clear:N __enumext_store_current_label_tl
3043 \int_compare:nNnTF { __enumext_keyans_pic_level_int } = { 1 }
3044 {
3045 \tl_put_right:Ne __enumext_store_current_label_tl { __enumext_label_vi_tl }
3046 }
3047 {
3048 \tl_put_right:Ne __enumext_store_current_label_tl { __enumext_label_v_tl }
3049 }

```

If the *optional argument* is present and the *save-sep* key is not empty, we save it.

```

3050 \tl_if_novalue:nF { #1 }
3051 {
3052 \tl_if_empty:NF __enumext_store_keyans_item_opt_sep_tl
3053 {
3054 \tl_put_right:Ne __enumext_store_current_label_tl
3055 {
3056 __enumext_store_keyans_item_opt_sep_tl
3057 }
3058 }
3059 \tl_put_right:Ne __enumext_store_current_label_tl { #1 }
3060 }
3061 __enumext_store_addto_prop:V __enumext_store_current_label_tl
3062 }

```

(End of definition for \\_\_enumext\_keyans\_addto\_prop:n.)

#### 13.33.2 The save-ref key for keyans, keyans\* and keyanspic

The “*internal label and ref*” system for the *keyans*, *keyans\** and *keyanspic* environments has *slight differences* with the one implemented for *\anskey* basically because in this environments the interest is in the current *⟨label⟩* for *\item\** and *\anspic\** with the *⟨contents⟩* of the *optional argument*. The mechanism defined here will allow to execute *\ref{⟨store name : position⟩}* and will return *1. (A)*.

\\_\_enumext\_keyans\_store\_ref:  
 \\_\_enumext\_keyans\_store\_ref\_aux:i  
 \\_\_enumext\_keyans\_store\_ref\_aux\_ii:

The function \\_\_enumext\_keyans\_store\_ref: handles the “*internal label and ref*” system used by the *save-ref* key for *\item\** and *\anspic\** commands. First we will create copies of the current *⟨labels⟩* and remove the dots “.” from them, we do not want to get double dots in references.

```

3063 \cs_new_protected:Nn __enumext_keyans_store_ref:
3064 {
3065 \bool_if:NT __enumext_store_ref_key_bool
3066 {
3067 \cs_set_protected:Npn __enumext_tmp:n ##1
3068 {

```

```

3069 \tl_set_eq:cc { __enumext_label_copy_##1_tl } { __enumext_label_##1_tl }
3070 \tl_reverse:c { __enumext_label_copy_##1_tl }
3071 \tl_remove_once:cn { __enumext_label_copy_##1_tl } { . }
3072 \tl_reverse:c { __enumext_label_copy_##1_tl }
3073 }
3074 \clist_map_inline:nn { i, v, vi, vii, viii } { __enumext_tmp:n {##1} }
3075 __enumext_keyans_store_ref_aux_i:
3076 }
3077 }

```

The auxiliary function `\__enumext_keyans_store_ref_aux_i:` set the variable `\__enumext_newlabel_arg_one_tl` which will contain  $\langle store\ name : position \rangle$  analyzing whether the environment in which they are executed is `enumext*` or `enumext`.

```

3078 \cs_new_protected:Nn __enumext_keyans_store_ref_aux_i:
3079 {
3080 \bool_if:NT \g__enumext_starred_bool
3081 {
3082 \tl_set_eq:NN __enumext_label_copy_i_tl __enumext_label_copy_vii_tl
3083 }
3084 \int_compare:nNnT { __enumext_keyans_pic_level_int } = { 1 }
3085 {
3086 \tl_put_right:Ne __enumext_newlabel_arg_two_tl
3087 { __enumext_label_copy_i_tl . __enumext_label_copy_vi_tl }
3088 }
3089 \int_compare:nNnT { __enumext_keyans_level_int } = { 1 }
3090 {
3091 \tl_put_right:Ne __enumext_newlabel_arg_two_tl
3092 { __enumext_label_copy_i_tl . __enumext_label_copy_v_tl }
3093 }
3094 \int_compare:nNnT { __enumext_keyans_level_h_int } = { 1 }
3095 {
3096 \tl_put_right:Ne __enumext_newlabel_arg_two_tl
3097 { __enumext_label_copy_i_tl . __enumext_label_copy_viii_tl }
3098 }
3099 \tl_put_right:Ne __enumext_newlabel_arg_one_tl
3100 {
3101 __enumext_store_name_tl \c_colon_str
3102 \int_eval:n { \prop_count:c { g__enumext_ __enumext_store_name_tl _prop } }
3103 }
3104 __enumext_keyans_store_ref_aux_ii:
3105 }

```

Now auxiliary function `\__enumext_keyans_store_ref_aux_ii:` save the result in the variable `\__enumext_write_aux_file_tl` and finally we write in the `.aux` file.

```

3106 \cs_new_protected:Nn __enumext_keyans_store_ref_aux_ii:
3107 {
3108 \tl_put_right:Ne __enumext_write_aux_file_tl
3109 {
3110 __enumext_newlabel:nn
3111 { \exp_not:V __enumext_newlabel_arg_one_tl }
3112 { __enumext_newlabel_arg_two_tl }
3113 }
3114 __enumext_write_aux_file_tl
3115 }

```

(End of definition for `\__enumext_keyans_store_ref:`, `\__enumext_keyans_store_ref_aux_i:`, and `\__enumext_keyans_store_ref_aux_ii:`.)

### 13.33.3 Storing content in sequence

`\__enumext_keyans_addto_seq:n`  
`\__enumext_keyans_addto_seq_link:`

The function `\__enumext_keyans_addto_seq:n` will pass the contents of the current  $\langle label \rangle$  `\__enumext_label_v_tl` for the `keyans` environment and the `\__enumext_label_vi_tl` for the `keyanspic` environment when using `\item*` and `\anspic*`, followed by the  $\langle contents \rangle$  of the *optional argument* of both commands to the `\__enumext_store_current_label_tl` variable to the sequence defined by the `save-ans` key.

```

3116 \cs_new_protected:Npn __enumext_keyans_addto_seq:n #1
3117 {
3118 \tl_clear:N __enumext_store_current_label_tl
3119 \int_compare:nNnTF { __enumext_keyans_pic_level_int } = { 1 }
3120 {
3121 \tl_put_right:Ne __enumext_store_current_label_tl { \item __enumext_label_vi_tl }
3122 }

```

```

3123 {
3124 \tl_put_right:Ne \l__enumext_store_current_label_tl { \item \l__enumext_label_v_tl }
3125 }
3126 \tl_if_novalue:nF { #1 }
3127 {
3128 \tl_if_empty:NF \l__enumext_store_keyans_item_opt_sep_tl
3129 {
3130 \tl_put_right:Ne \l__enumext_store_current_label_tl
3131 {
3132 \l__enumext_store_keyans_item_opt_sep_tl
3133 }
3134 }
3135 \tl_put_right:Ne \l__enumext_store_current_label_tl { #1 }
3136 }
3137 __enumext_keyans_addto_seq_link:
3138 }

```

Checks if the `save-ref` key is active along with the `hyperref` package load, if both conditions are met, it will create the `\hyperlink` and then store using the `\__enumext_store_addto_seq:V` function. Finally, copy the contents of the variable `\l__enumext_store_current_label_tl` into the global variable `\g__enumext_check_ans_item_tl` to be used by the function `\__enumext_check_starred_cmd:n` and increment the value of the integer variable `\g__enumext_item_anskey_int` handled by the `check-ans` key.

```

3139 \cs_new_protected:Nn __enumext_keyans_addto_seq_link:
3140 {
3141 \bool_lazy_and:nnT
3142 { \bool_if_p:N \l__enumext_store_ref_key_bool }
3143 { \bool_if_p:N \l__enumext_hyperref_bool }
3144 {
3145 \tl_put_right:Ne \l__enumext_store_current_label_tl
3146 {
3147 \hfill \exp_not:N \hyperlink
3148 {
3149 \exp_not:V \l__enumext_newlabel_arg_one_tl
3150 }
3151 { \exp_not:V \l__enumext_mark_ref_sym_tl }
3152 }
3153 }
3154 __enumext_store_addto_seq:V \l__enumext_store_current_label_tl
3155 \bool_if:NT \l__enumext_check_answers_bool
3156 {
3157 \int_gincr:N \g__enumext_item_anskey_int
3158 }
3159 }

```

(End of definition for `\__enumext_keyans_addto_seq:n` and `\__enumext_keyans_addto_seq_link:.`)

### 13.33.4 The `show-ans` and `show-pos` keys for `keyans` and `keyanspic`

The code is very similar to the `\anskey` code, but, if I change the order of the operations the counter off `\label` are incorrect.

```

__enumext_keyans_show_left:n
__enumext_keyans_show_ans:
__enumext_keyans_show_pos:
__enumext_keyans_show_item_opt:

```

Common function to show *starred commands* `\item*` and `\position` of stored content in *prop list* for `keyans` and `keyanspic`. Need add `1` to `\g__enumext_⟨store name⟩_prop` for `show-pos` key.

```

3160 \cs_new_protected:Npn __enumext_keyans_show_left:n #1
3161 {
3162 \tl_if_novalue:nF { #1 }
3163 {
3164 \tl_set:Ne \l__enumext_store_current_opt_arg_tl { #1 }
3165 }
3166 \bool_if:NT \l__enumext_show_answer_bool
3167 {
3168 __enumext_keyans_show_ans:
3169 }
3170 \bool_if:NT \l__enumext_show_position_bool
3171 {
3172 __enumext_keyans_show_pos:
3173 }
3174 }
3175 \cs_new_protected:Nn __enumext_keyans_show_item_opt:
3176 {
3177 \tl_if_empty:NF \l__enumext_store_current_opt_arg_tl

```

```

3178 {
3179 \bool_lazy_or:nnT
3180 { \bool_if_p:N \l__enumext_show_answer_bool }
3181 { \bool_if_p:N \l__enumext_show_position_bool }
3182 {
3183 __enumext_keyans_wrapper_opt:n { \l__enumext_store_current_opt_arg_tl } \c_space_tl
3184 }
3185 }
3186 }
3187 \cs_new_protected:Nn __enumext_keyans_show_ans:
3188 {
3189 \bool_if:NT \l__enumext_starred_bool
3190 {
3191 \dim_set_eq:NN \l__enumext_labelwidth_i_dim \l__enumext_labelwidth_vii_dim
3192 \dim_set_eq:NN \l__enumext_labelsep_i_dim \l__enumext_labelsep_vii_dim
3193 }
3194 \tl_put_left:Nn \l__enumext_label_v_tl
3195 {
3196 __enumext_print_keyans_box:NN
3197 \l__enumext_labelwidth_i_dim \l__enumext_labelsep_i_dim
3198 }
3199 }
3200 \cs_new_protected:Nn __enumext_keyans_show_pos:
3201 {
3202 \bool_if:NT \l__enumext_starred_bool
3203 {
3204 \dim_set_eq:NN \l__enumext_labelwidth_i_dim \l__enumext_labelwidth_vii_dim
3205 \dim_set_eq:NN \l__enumext_labelsep_i_dim \l__enumext_labelsep_vii_dim
3206 }
3207 \int_compare:nNnTF { \l__enumext_keyans_pic_level_int } = { 1 }
3208 {
3209 \tl_set:Ne \l__enumext_mark_answer_sym_tl
3210 {
3211 \group_begin:
3212 \exp_not:N \normalfont
3213 \exp_not:N \footnotesize [\int_eval:n
3214 {
3215 \prop_count:c { g__enumext_ \l__enumext_store_name_tl _prop }
3216 }
3217]
3218 \group_end:
3219 }
3220 }
3221 {
3222 \tl_set:Ne \l__enumext_mark_answer_sym_tl
3223 {
3224 \group_begin:
3225 \exp_not:N \normalfont
3226 \exp_not:N \footnotesize [\int_eval:n
3227 {
3228 \prop_count:c { g__enumext_ \l__enumext_store_name_tl _prop } + 1
3229 }
3230]
3231 \group_end:
3232 }
3233 }
3234 \tl_put_left:Nn \l__enumext_label_v_tl
3235 {
3236 __enumext_print_keyans_box:NN
3237 \l__enumext_labelwidth_i_dim \l__enumext_labelsep_i_dim
3238 }
3239 }

```

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

### 13.34 Redefining `\item` and `\makelabel` in `enumext`

Redefining the `\item` command is not as simple as I thought. This command works in conjunction with the `\makelabel` command so I have to redefine both of them, in addition to this, we will have to use a couple of *global* variables to pass the values from one command to the other.

When *labeling* PDF is active `\makelabel` is redefined as `\hss #1` and the only way to get the `align` key to work correctly is to redefine `\makelabel` using `\makebox`. The best way to implement this is to use the

conditional command `\IfDocumentMetadataTF` to force this redefinition and the dedicated `mode-box` key to manually activate it by the user.

The `\item` and `\item[⟨custom⟩]` commands work in the usual way on `enumext` and we will add `\item*`, `\item*[⟨symbol⟩]` and `\item*[⟨symbol⟩][⟨offset⟩]`.

`\__enumext_default_item:n`

First we will see if the *optional argument* is present, if it is NOT present we will check the state of the variable `\l__enumext_check_answers_bool` set by the key `no-store`, set the boolean variable `\l__enumext_wrap_label_X_bool` to “true” for the key `wrap-label` and execute `\__enumext_item_std:w` and the key `itemindent`, otherwise we will check the state of the boolean variable `\l__enumext_wrap_label_opt_X_bool` set by the key `wrap-label*` and execute `\__enumext_item_std:w` with the *optional argument* and the key `itemindent`.

```

3240 \cs_new_protected:Npn __enumext_default_item:n #1
3241 {
3242 \tl_if_novalue:nTF {#1}
3243 {
3244 \bool_if:NT \l__enumext_check_answers_bool
3245 {
3246 \int_gincr:N \g__enumext_item_number_int
3247 \bool_set_true:N \l__enumext_item_number_bool
3248 }
3249 \bool_set_true:c { l__enumext_wrap_label_ __enumext_level: _bool }
3250 __enumext_item_std:w \tl_use:c { l__enumext_fake_item_indent_ __enumext_level: _tl }
3251 }
3252 {
3253 \bool_set_eq:cc
3254 { l__enumext_wrap_label_ __enumext_level: _bool }
3255 { l__enumext_wrap_label_opt_ __enumext_level: _bool }
3256 __enumext_item_std:w [#1] \tl_use:c { l__enumext_fake_item_indent_ __enumext_level: _tl }
3257 }
3258 }

```

(End of definition for `\__enumext_default_item:n`.)

`\__enumext_starred_item:nn`

`\__enumext_item_star_exec:`

The `\item*`, `\item*[⟨symbol⟩]` and `\item*[⟨symbol⟩][⟨offset⟩]` works like the *numbered* `\item`, but placing a `⟨symbol⟩` to the “left” of the `⟨label⟩` separated from it by the value the second *optional argument* `⟨offset⟩`.

**#1:** `\l__enumext_item_symbol_X_tl`

**#2:** `\l__enumext_item_symbol_sep_X_dim`

First we will make a copy of `\l__enumext_item_symbol_X_tl` which is set by the key `item-sym*` or passed as “first” *optional argument* in the global variable `\g__enumext_item_symbol_aux_tl`, followed by setting the variable `\l__enumext_item_symbol_sep_X_dim` set by the key `item-pos*` or by the “second” *optional argument*, then we will see the state of the variable `\l__enumext_check_answers_bool` set by the key `no-store`, set the boolean variable `\l__enumext_wrap_label_X_bool` to “true” for the key `wrap-label` and execute `\__enumext_item_std:w` and the key `itemindent`.

```

3259 \cs_new_protected:Npn __enumext_starred_item:nn #1 #2
3260 {
3261 \tl_if_novalue:nTF {#1}
3262 {
3263 \tl_gset_eq:Nc
3264 \g__enumext_item_symbol_aux_tl { l__enumext_item_symbol_ __enumext_level: _tl }
3265 }
3266 {
3267 \tl_gset:Nn \g__enumext_item_symbol_aux_tl {#1}
3268 }
3269 \tl_if_novalue:nTF {#2}
3270 {
3271 \dim_set_eq:cc
3272 { l__enumext_item_symbol_sep_ __enumext_level: _dim }
3273 { l__enumext_labelsep_ __enumext_level: _dim }
3274 }
3275 {
3276 \dim_set:cn { l__enumext_item_symbol_sep_ __enumext_level: _dim } {#2}
3277 }
3278 \bool_if:NT \l__enumext_check_answers_bool
3279 {
3280 \int_gincr:N \g__enumext_item_number_int
3281 \bool_set_true:N \l__enumext_item_number_bool
3282 }
3283 \bool_set_true:c { l__enumext_wrap_label_ __enumext_level: _bool }

```



```

3284 __enumext_item_std:w \tl_use:c { l__enumext_fake_item_indent_ __enumext_level: _tl }
3285 }

```

The function `\__enumext_item_star_exec:` will be responsible for executing `\item*` for the `enumext` environment.

```

3286 \cs_new_protected:Nn __enumext_item_star_exec:
3287 {
3288 \tl_if_empty:cF { l__enumext_item_symbol_ __enumext_level: _tl }
3289 {
3290 \mode_leave_vertical:
3291 \skip_horizontal:n { -\dim_use:c { l__enumext_item_symbol_sep_ __enumext_level: _dim } }
3292 \hbox_overlap_left:n { \g__enumext_item_symbol_aux_tl }
3293 \skip_horizontal:n { \dim_use:c { l__enumext_item_symbol_sep_ __enumext_level: _dim } }
3294 }
3295 }

```

(End of definition for `\__enumext_starred_item:nn` and `\__enumext_item_star_exec:`.)

`\__enumext_redefine_item:` The function `\__enumext_redefine_item:` will redefine the `\item` command in the `enumext` environment adding `\item*`. This function are passed to `\__enumext_list_arg_two_X:` used in the definition of the `enumext` environment (§13.39).

```

3296 \cs_new_protected:Nn __enumext_redefine_item:
3297 {
3298 \RenewDocumentCommand \item { s o o }
3299 {
3300 \bool_if:nTF {##1}
3301 {
3302 __enumext_starred_item:nn {##2} {##3}
3303 }
3304 { __enumext_default_item:n {##2} }
3305 }
3306 }

```

(End of definition for `\__enumext_redefine_item:`.)

`\__enumext_make_label:` The function `\__enumext_make_label:` redefine `\makeLabel` for the keys `mode-box`, `align`, `font`, `wrap-label`, `wrap-label*` and `\item*` for `enumext` environment. This function are passed to `\__enumext_list_arg_two_X:` used in the definition of the `enumext` environment (§13.39).

```

3307 \cs_new_protected:Nn __enumext_make_label:
3308 {
3309 \IfDocumentMetadataTF
3310 {
3311 __enumext_make_label_box:
3312 }
3313 {
3314 \bool_if:NTF \l__enumext_mode_box_bool
3315 {
3316 __enumext_make_label_box:
3317 }
3318 {
3319 __enumext_make_label_std:
3320 }
3321 }
3322 }

```

Standard definition when `\DocumentMetadata` is not active.

```

3323 \cs_new_protected:Nn __enumext_make_label_std:
3324 {
3325 \RenewDocumentCommand \makeLabel { m }
3326 {
3327 \tl_use:c { l__enumext_label_fill_left_ __enumext_level: _tl }
3328 \tl_use:c { l__enumext_label_font_style_ __enumext_level: _tl }
3329 \bool_if:cTF { l__enumext_wrap_label_ __enumext_level: _bool }
3330 {
3331 __enumext_item_star_exec:
3332 \use:c { __enumext_wrapper_label_ __enumext_level: :n } { ##1 }
3333 }
3334 { ##1 }
3335 \tl_use:c { l__enumext_label_fill_right_ __enumext_level: _tl }
3336 \tl_gclear:N \g__enumext_item_symbol_aux_tl
3337 }
3338 }

```

Definition using `\makebox` when `\DocumentMetadata` is active or `mode-box` is active.

- Here it is necessary to use `\strut\smash` to maintain text *alignment* in case the user wants to use `\labelbx` for example. In my experiments with *mimicking* the `description` environment it was the only way out and it seems to have no adverse effects and may serve in the future as a basis for a more generic `list` environment package than `enumext`.

```

3339 \cs_new_protected:Nn __enumext_make_label_box:
3340 {
3341 \RenewDocumentCommand \make_label { m }
3342 {
3343 \strut\smash
3344 {
3345 \makebox
3346 [\dim_use:c { l__enumext_labelwidth_ __enumext_level: _dim }]
3347 [\str_use:c { l__enumext_align_label_pos_ __enumext_level: _str }]
3348 {
3349 \tl_use:c { l__enumext_label_font_style_ __enumext_level: _tl }
3350 \bool_if:cTF { l__enumext_wrap_label_ __enumext_level: _bool }
3351 {
3352 __enumext_item_star_exec:
3353 \use:c { __enumext_wrapper_label_ __enumext_level: :n } { ##1 }
3354 }
3355 { ##1 }
3356 \tl_gclear:N \g__enumext_item_symbol_aux_tl
3357 }
3358 } % close smash
3359 }
3360 }

```

(End of definition for `\__enumext_make_label:`, `\__enumext_make_label_std:`, and `\__enumext_make_label_box:`.)

### 13.35 Setting `item-sym*` and `item-pos*` keys

In order to have a cleaner implementation of `\item*` for the `enumext` and `enumext*` environments it is best to define a couple of keys that allow us to control and set by default the `<symbol>` and its `<offset>`.

`item-sym*` Define and set `item-sym*` and `item-pos*` keys for `enumext` and `enumext*`.

```

item-pos*
3361 \cs_set_protected:Npn __enumext_tmp:nn #1 #2
3362 {
3363 \keys_define:nn { enumext / #1 }
3364 {
3365 item-sym* .tl_set:c = { l__enumext_item_symbol_#2_tl },
3366 item-sym* .value_required:n = true,
3367 item-sym* .initial:n = {\textasteriskcentered},
3368 item-pos* .dim_set:c = { l__enumext_item_symbol_sep_#2_dim },
3369 item-pos* .value_required:n = true,
3370 }
3371 }
3372 \clist_map_inline:nn
3373 {
3374 {level-1}{i}, {level-2}{ii}, {level-3}{iii}, {level-4}{iv}, {enumext*}{vii}
3375 }
3376 { __enumext_tmp:nn #1 }

```

(End of definition for `item-sym*` and `item-pos*`.)

### 13.36 Handling unknown keys

At this point in the code I already know that I will not add more `<keys>` and since I have already been quite *paranoid and restrictive* with the definitions of environments and commands, the only thing left to do is do it with the `<keys>` (you have to be consistent in life).

#### 13.36.1 Handling unknown keys for `keyans`, `keyans*` and `keyanspic`

`unknown` Define and set `unknown` key for `keyans`, `keyans*` and `keyanspic` environments.

```

__enumext_keyans_unknown_keys:n
__enumext_keyans_unknown_keys:nn
3377 \cs_set_protected:Npn __enumext_tmp:n #1
3378 {
3379 \keys_define:nn { enumext / #1 }
3380 {
3381 unknown .code:n = { __enumext_keyans_unknown_keys:n {##1} }
3382 }
3383 }
3384 \clist_map_inline:nn { keyans, keyans*, keyanspic } { __enumext_tmp:n {#1} }

```

Internal functions for handling `unknown` key.

```

3385 \cs_new_protected:Npn __enumext_keyans_unknown_keys:n #1
3386 {
3387 \exp_args:NV __enumext_keyans_unknown_keys:nn \l_keys_key_str {#1}
3388 }
3389 \cs_new_protected:Npn __enumext_keyans_unknown_keys:nn #1#2
3390 {
3391 \tl_if_blank:nTF {#2}
3392 {
3393 \msg_error:nnn { enumext } { keyans-unknown-key } {#1}
3394 }
3395 {
3396 \msg_error:nnnn { enumext } { keyans-unknown-key-value } {#1} {#2}
3397 }
3398 }

```

(End of definition for `unknown`, `\__enumext_keyans_unknown_keys:n`, and `\__enumext_keyans_unknown_keys:nn`.)

### 13.36.2 Handling unknown keys for `enumext*`

Define and set `unknown` key for `enumext*` environment.

```

unknown
__enumext_starred_unknown_keys:n
__enumext_starred_unknown_keys:nn
3399 \keys_define:nn { enumext / enumext* }
3400 {
3401 unknown .code:n = { __enumext_starred_unknown_keys:n {#1} }
3402 }

```

Internal functions for handling `unknown` key.

```

3403 \cs_new_protected:Npn __enumext_starred_unknown_keys:n #1
3404 {
3405 \exp_args:NV __enumext_starred_unknown_keys:nn \l_keys_key_str {#1}
3406 }
3407 \cs_new_protected:Npn __enumext_starred_unknown_keys:nn #1#2
3408 {
3409 \tl_if_blank:nTF {#2}
3410 {
3411 \msg_error:nnn { enumext } { starred-unknown-key } {#1}
3412 }
3413 {
3414 \msg_error:nnnn { enumext } { starred-unknown-key-value } {#1} {#2}
3415 }
3416 }

```

(End of definition for `unknown`, `\__enumext_starred_unknown_keys:n`, and `\__enumext_starred_unknown_keys:nn`.)

### 13.36.3 Handling unknown keys for `enumext`

Defines and set the key `unknown` for `enumext` environment.

```

unknown
__enumext_standar_unknown_keys:n
__enumext_standar_unknown_keys:nn
3417 \cs_set_protected:Npn __enumext_tmp:n #1
3418 {
3419 \keys_define:nn { enumext / #1 }
3420 {
3421 unknown .code:n = { __enumext_standar_unknown_keys:n {##1} }
3422 }
3423 }
3424 \clist_map_inline:nn { level-1,level-2,level-3,level-4 } { __enumext_tmp:n {#1} }

```

Internal functions for handling `unknown` key.

```

3425 \cs_new_protected:Npn __enumext_standar_unknown_keys:n #1
3426 {
3427 \exp_args:NV __enumext_standar_unknown_keys:nn \l_keys_key_str {#1}
3428 }
3429 \cs_new_protected:Npn __enumext_standar_unknown_keys:nn #1#2
3430 {
3431 \tl_if_blank:nTF {#2}
3432 {
3433 \msg_error:nnn { enumext } { standar-unknown-key } {#1}
3434 }
3435 {
3436 \msg_error:nnnn { enumext } { standar-unknown-key-value } {#1} {#2}
3437 }
3438 }

```

(End of definition for `unknown`, `\__enumext_standar_unknown_keys:n`, and `\__enumext_standar_unknown_keys:nn`.)

### 13.37 Redefining \item and \makeLabel in keyans

The `\item` and `\item[⟨custom⟩]` commands work in the usual way in `keyans`, but the `\item*` and `\item*[⟨content⟩]` commands *store* the current `⟨label⟩` next to the `⟨content⟩` if it is present in the *sequence* and *prop list* defined by `save-ans` key.

`\__enumext_keyans_default_item:n`

The function `\__enumext_keyans_default_item:n` executes the original behavior of the `\item` along with the keys `wrap-label`, `wrap-label*` and `itemindent`.

```

3439 \cs_new_protected:Npn __enumext_keyans_default_item:n #1
3440 {
3441 \tl_if_novalue:nTF { #1 }
3442 {
3443 \bool_set_true:N __enumext_wrap_label_v_bool
3444 __enumext_item_std:w \tl_use:N __enumext_fake_item_indent_v_tl
3445 }
3446 {
3447 \bool_set_eq:NN __enumext_wrap_label_v_bool __enumext_wrap_label_opt_v_bool
3448 __enumext_item_std:w [#1] \tl_use:N __enumext_fake_item_indent_v_tl
3449 }
3450 }
```

(End of definition for `\__enumext_keyans_default_item:n`.)

`\__enumext_keyans_starred_item:n`

The function `\__enumext_keyans_starred_item:n` which will make a temporary copy of the current `⟨label⟩`, execute the `show-ans` or `show-pos` keys using the function `\__enumext_keyans_show_left:n` and will display the `⟨contents⟩` of that item using the internal copy `\__enumext_item_std:w`, this is necessary to prevent incrementing the current “counter” of the original `⟨label⟩`, followed by this it will execute function `\__enumext_keyans_show_item_opt:` handled by `wrap-opt` key.

```

3451 \cs_new_protected:Npn __enumext_keyans_starred_item:n #1
3452 {
3453 \tl_set_eq:NN __enumext_store_current_label_tmp_tl __enumext_label_v_tl
3454 __enumext_keyans_show_left:n { #1 }
3455 \bool_set_true:N __enumext_wrap_label_v_bool
3456 __enumext_item_std:w \tl_use:N __enumext_fake_item_indent_v_tl
3457 __enumext_keyans_show_item_opt:
```

Recover the original value of the current `⟨label⟩` and *store* it first in the *prop list* (including the *optional argument*), run the internal “*label and ref*” system if the `save-ref` key is active, *store* it in the *sequence* and finally increments `\g__enumext_check_starred_cmd_int` for internal check system.

```

3458 \tl_set_eq:NN __enumext_label_v_tl __enumext_store_current_label_tmp_tl
3459 __enumext_keyans_addto_prop:n { #1 }
3460 __enumext_keyans_store_ref:
3461 __enumext_keyans_addto_seq:n { #1 }
3462 \int_gincr:N \g__enumext_check_starred_cmd_int
3463 }
```

(End of definition for `\__enumext_keyans_starred_item:n`.)

`\item*`

`\__enumext_keyans_redefine_item:`

The function `\__enumext_keyans_redefine_item:` is responsible for adding the *starred argument* and *optional argument* by the `\__enumext_list_arg_two_v:` function in the definition of the `keyans` environment. Here we need to use `\peek_remove_spaces:n` to prevent an unwanted space when using `\item*` in conjunction with the `itemindent` key. This function are passed to `\__enumext_list_arg_two_v:` used in the definition of the `keyans` environment (§13.38).

```

3464 \cs_new_protected:Nn __enumext_keyans_redefine_item:
3465 {
3466 \RenewDocumentCommand \item { s o }
3467 {
3468 \bool_if:nTF {##1}
3469 {
3470 \int_incr:N __enumext_item_star_exec_int % increment
3471 \peek_remove_spaces:n
3472 {
3473 __enumext_keyans_starred_item:n {##2}
3474 }
3475 }
3476 {
3477 \int_zero:N __enumext_item_star_exec_int % zero
3478 __enumext_keyans_default_item:n {##2}
3479 }
3480 }
3481 }
```

(End of definition for `\item*` and `\__enumext_keyans_redefine_item:`. This function is documented on page 15.)

```
__enumext_keyans_make_label:
__enumext_keyans_wrapper_label:n
__enumext_keyans_make_label_std:
__enumext_keyans_make_label_box:
```

The function `\__enumext_keyans_make_label:` redefine `\makeLabel` for the keys `mode-box`, `align`, `font`, `wrap-label`, `wrap-label*` and `\item*` for `keyans` environment. This function are passed to `\__enumext_list_arg_two_v:` used in the definition of the `keyans` environment (§13.38).

```
3482 \cs_new_protected:Nn __enumext_keyans_make_label:
3483 {
3484 \IfDocumentMetadataTF
3485 {
3486 __enumext_keyans_make_label_box:
3487 }
3488 {
3489 \bool_if:NTF \l__enumext_mode_box_bool
3490 {
3491 __enumext_keyans_make_label_box:
3492 }
3493 {
3494 __enumext_keyans_make_label_std:
3495 }
3496 }
3497 }
```

We added conditionals to the `\__enumext_keyans_wrapper_label:n` function to handle the keys `wrap-key`, `wrap-label` and `wrap-label*`.

```
3498 \cs_new_protected:Npn __enumext_keyans_wrapper_label:n #1
3499 {
3500 \bool_lazy_all:nT
3501 {
3502 { \bool_if_p:N \l__enumext_wrap_label_v_bool }
3503 { \bool_if_p:N \l__enumext_show_answer_bool }
3504 { \int_compare_p:nNn { \l__enumext_item_star_exec_int } = { 1 } }
3505 { \cs_if_exist_p:N __enumext_keyans_wrapper_item:n }
3506 }
3507 {
3508 \cs_set_eq:NN __enumext_wrapper_label_v:n __enumext_keyans_wrapper_item:n
3509 }
3510 \bool_if:NTF \l__enumext_wrap_label_v_bool
3511 {
3512 __enumext_wrapper_label_v:n { #1 }
3513 }
3514 { #1 }
3515 }
```

Standard definition when `\DocumentMetadata` is not active.

```
3516 \cs_new_protected:Nn __enumext_keyans_make_label_std:
3517 {
3518 \RenewDocumentCommand \makeLabel { m }
3519 {
3520 \tl_use:N \l__enumext_label_fill_left_v_tl
3521 \tl_use:N \l__enumext_label_font_style_v_tl
3522 __enumext_keyans_wrapper_label:n { ##1 }
3523 \tl_use:N \l__enumext_label_fill_right_v_tl
3524 }
3525 }
```

Definition using `\makebox` when `\DocumentMetadata` is active or `mode-box` is active.

```
3526 \cs_new_protected:Nn __enumext_keyans_make_label_box:
3527 {
3528 \RenewDocumentCommand \makeLabel { m }
3529 {
3530 \strut\smash
3531 {
3532 \makebox[\l__enumext_labelwidth_v_dim][\l__enumext_align_label_pos_v_str]
3533 {
3534 \tl_use:N \l__enumext_label_font_style_v_tl
3535 __enumext_keyans_wrapper_label:n { ##1 }
3536 }
3537 }
3538 }
3539 }
```

(End of definition for `\__enumext_keyans_make_label:` and others.)

### 13.38 Second argument of the lists

At this point of the code we have already programmed most the necessary tools to create a custom `list` environment, remember that the function `\__enumext_start_list:nn` takes two arguments, the first one we have ready, the second one we will define for all the levels of the environment `enumext` and the environment `keyans`.

#### 13.38.1 Calculation of `\leftmargin` and `\itemindent`

Consider the figure 9 where the default margins (on the left) of a list are represented.

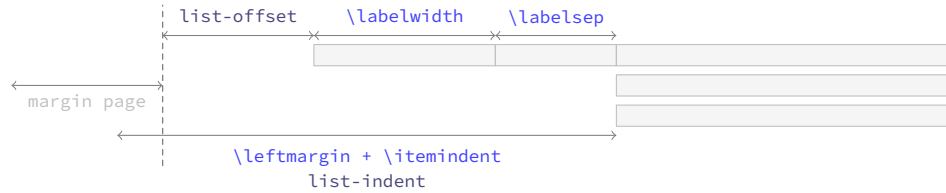


Figure 9: Representation of standard horizontal lengths in `list` environment.

The idea is to have control over these margins so that our list does not overlap the left margin of the page. The key relationship is that the right edge of the `\labelsep` equals the right edge of the `\itemindent`, so that the left edge of the *label box* is at `\leftmargin + \itemindent` minus `\labelwidth + \labelsep`. Thus, the handling of the margins by the package will be as shown in the figure 10.

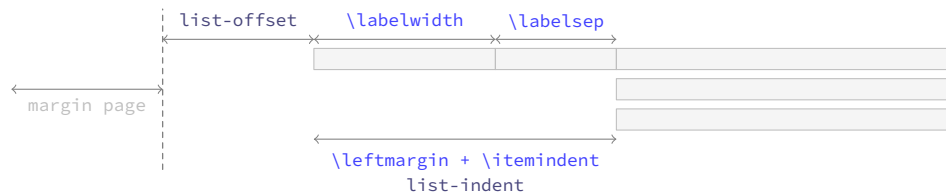


Figure 10: Representation of horizontal lengths concept in list in `enumext`.

Where the default values will look like in the figure 11.

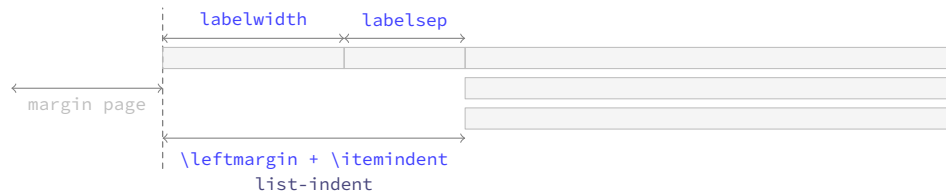


Figure 11: Default horizontal lengths in `enumext`.

```
__enumext_calc_hspace:NNNNNNN
__enumext_calc_hspace:ccccccc
```

The function `\__enumext_calc_hspace:NNNNNNN` takes seven arguments to be able to determine horizontal spaces for all list environment:

```
#1: \l__enumext_labelwidth_X_dim #2: \l__enumext_labelsep_X_dim
#3: \l__enumext_listoffset_X_dim #4: \l__enumext_leftmargin_tmp_X_dim
#5: \l__enumext_leftmargin_X_dim #6: \l__enumext_itemindent_X_dim
#7: \l__enumext_leftmargin_tmp_X_bool
```

And returns the “adjusted” values of `\leftmargin` and `\itemindent`.

This function is passed to `\__enumext_list_arg_two_X:` which is used in the definition of the `enumext` and `keyans` environments (§13.38).

```
3540 \cs_new_protected:Npn __enumext_calc_hspace:NNNNNNN #1 #2 #3 #4 #5 #6 #7
3541 {
3542 \dim_compare:nNt { #1 } < { \c_zero_dim }
3543 {
3544 \msg_warning:nnnV { enumext } { width-non-positive } { labelwidth } { #1 }
3545 \dim_set:Nn #1 { \dim_abs:n { #1 } }
3546 }
3547 \dim_compare:nNt { #2 } < { \c_zero_dim }
3548 {
3549 \msg_warning:nnnV { enumext } { width-negative } { labelsep } { #2 }
3550 \dim_set:Nn #2 { \dim_abs:n { #2 } }
3551 }
}
```

If no value has been passed to the `labelwidth` and `labelsep` keys we set the default values for `\l__enumext_leftmargin_tmp_X_dim`.

```
3552 \bool_if:nF #7 { \dim_set:Nn #4 { #1 + #2 } }
```

We now analyze the cases and set the values for `\leftmargin` and `\itemindent`.

```

3553 \dim_compare:nNnTF { #4 } < { \c_zero_dim }
3554 {
3555 \dim_set:Nn #6 { #1 + #2 - #4 }
3556 \dim_set:Nn #5 { #1 + #2 + #3 - #6 }
3557 }
3558 {
3559 \dim_compare:nNnT { #4 } = { #1 + #2 }
3560 { \dim_set:Nn #6 { \c_zero_dim } }
3561 \dim_compare:nNnT { #4 } < { #1 + #2 }
3562 { \dim_set:Nn #6 { #1 + #2 - #4 } }
3563 \dim_compare:nNnT { #4 } > { #1 + #2 }
3564 {
3565 \dim_set:Nn #6 { -#1 - #2 + #4 }
3566 \dim_set:Nn #6 { #6*-1 }
3567 }
3568 \dim_set:Nn #5 { #1 + #2 + #3 - #6 }
3569 }
3570 }
3571 \cs_generate_variant:Nn __enumext_calc_hspace:NNNNNNN { cccccc }

```

(End of definition for `\__enumext_calc_hspace:NNNNNNN`.)

### 13.38.2 Setting second argument of the lists

We will “not set” `\leftmargini`, `\leftmarginii`, `\leftmarginiii` or `\leftmarginiv`, in this case, we will directly set the parameters for vertical and horizontal list spacing per level.

```

__enumext_list_arg_two_i:
__enumext_list_arg_two_ii:
__enumext_list_arg_two_iii:
__enumext_list_arg_two_iv:
__enumext_list_arg_two_v:
3572 \cs_set_protected:Npn __enumext_tmp:n #1
3573 {
3574 \cs_new_protected:cpn { __enumext_list_arg_two_#1: }
3575 {
3576 __enumext_calc_hspace:ccccc
3577 { __enumext_labelwidth_#1_dim } { __enumext_labelsep_#1_dim }
3578 { __enumext_listoffset_#1_dim } { __enumext_leftmargin_tmp_#1_dim }
3579 { __enumext_leftmargin_#1_dim } { __enumext_itemindent_#1_dim }
3580 { __enumext_leftmargin_tmp_#1_bool }
3581 \clist_map_inline:nn
3582 { labelsep, labelwidth, itemindent, leftmargin, rightmargin, listparindent }
3583 { \dim_set_eq:cc {###1} { __enumext_###1_#1_dim } }
3584 \clist_map_inline:nn { topsep, parsep, partopsep, itemsep }
3585 { \skip_set_eq:cc {###1} { __enumext_###1_#1_skip } }
3586 \usecounter { enumX#1 }
3587 \setcounter { enumX#1 } { \int_eval:n { \int_use:c { __enumext_start_#1_int } - 1 } }
3588 \str_if_eq:nnTF {#1} { v }
3589 {
3590 __enumext_keyans_redefine_item:
3591 __enumext_keyans_make_label:
3592 __enumext_keyans_ref:
3593 __enumext_keyans_fake_item_indent:
3594 \bool_if:cT { __enumext_show_length_#1_bool }
3595 {
3596 \msg_term:nnnn { enumext } { list-lengths-not-nested } { v } { keyans }
3597 }
3598 }
3599 {
3600 __enumext_redefine_item:
3601 __enumext_make_label:
3602 __enumext_standar_ref:
3603 __enumext_fake_item_indent:
3604 \bool_if:cT { __enumext_show_length_#1_bool }
3605 {
3606 \msg_term:nnne { enumext } { list-lengths } {#1}
3607 { \int_use:N __enumext_level_int }
3608 }
3609 }
3610 }
3611 }
3612 \clist_map_inline:nn { i, ii, iii, iv, v } { __enumext_tmp:n {#1} }

```

(End of definition for `\__enumext_list_arg_two_i: and others`.)



```

__enumext_list_arg_two_vii:
__enumext_list_arg_two_viii:

```

For the horizontal environments `enumext*` and `keyans*` the implementation is similar, but, the value of `\partopsep` is always `0pt`. At this point we will modify the `parsep` key to make it take the value of the `itemsep` key and later, in the environment definition, we will modify `parindent` to make it set the value of `listparindent` and `parsep` to set the value of `\parskip` locally.

```

3613 \cs_set_protected:Npn __enumext_tmp:n #1
3614 {
3615 \cs_new_protected:cpn { __enumext_list_arg_two_#1: }
3616 {
3617 \bool_set_true:c { l__enumext_leftmargin_tmp_#1_bool }
3618 \dim_zero:c { l__enumext_leftmargin_tmp_#1_dim }
3619 __enumext_calc_hspace:ccccc
3620 { l__enumext_labelwidth_#1_dim } { l__enumext_labelsep_#1_dim }
3621 { l__enumext_listoffset_#1_dim } { l__enumext_leftmargin_tmp_#1_dim }
3622 { l__enumext_leftmargin_#1_dim } { l__enumext_itemindent_#1_dim }
3623 { l__enumext_leftmargin_tmp_#1_bool }
3624 \clist_map_inline:nn
3625 { labelsep, labelwidth, itemindent, leftmargin, rightmargin, listparindent }
3626 { \dim_set_eq:cc {####1} { l__enumext_####1_#1_dim } }
3627 \clist_map_inline:nn { topsep, parsep, partopsep, itemsep }
3628 { \skip_set_eq:cc {####1} { l__enumext_####1_#1_skip } }
3629 \skip_set_eq:Nc \parsep { l__enumext_itemsep_#1_skip }
3630 \skip_zero:N \partopsep
3631 \usecounter { enumX#1 }
3632 \setcounter { enumX#1 } { \int_eval:n { \int_use:c { l__enumext_start_#1_int } - 1 } }
3633 __enumext_starred_ref:
3634 \str_if_eq:nnTF {#1} { vii }
3635 {
3636 __enumext_fake_item_indent_vii:
3637 \bool_if:cT { l__enumext_show_length_vii_bool }
3638 { \msg_term:nnnn { enumext } { list-lengths-not-nested } { vii } { enumext* } }
3639 }
3640 {
3641 __enumext_fake_item_indent_viii:
3642 \bool_if:cT { l__enumext_show_length_#1_bool }
3643 { \msg_term:nnnn { enumext } { list-lengths-not-nested } { #1 } { keyans* } }
3644 }
3645 }
3646 }
3647 \clist_map_inline:nn { vii, viii } { __enumext_tmp:n {#1} }

```

(End of definition for `\__enumext_list_arg_two_vii:` and `\__enumext_list_arg_two_viii:`)

### 13.39 The environment `enumext`

```
__enumext_safe_exec:
```

The `\__enumext_safe_exec:` function first call the function `\__enumext_is_not_nested:` which sets `\g__enumext_standar_bool` to “true” if we are NOT nested within `enumext*`, then call the function `\__enumext_internal_mini_page:` to create the environment `__enumext_mini_page`, we will increment `\l__enumext_level_int` to restrict nesting of the environment, set `\l__enumext_standar_bool` to “true” and finally call the function `\__enumext_is_on_first_level:` which sets `\l__enumext_standar_first_bool` to “true” only if the environment is NOT nested and we are at the “first level”.

```

3648 \cs_new_protected:Nn __enumext_safe_exec:
3649 {
3650 __enumext_is_not_nested:
3651 __enumext_internal_mini_page:
3652 \int_incr:N \l__enumext_level_int
3653 \int_compare:nNnT { \l__enumext_level_int } > { 4 }
3654 { \msg_fatal:nn { enumext } { list-too-deep } }
3655 \bool_set_true:N \l__enumext_standar_bool
3656 \bool_set_false:N \l__enumext_starred_bool
3657 __enumext_is_on_first_level:
3658 }

```

(End of definition for `\__enumext_safe_exec:`)

```
__enumext_parse_keys:n
```

The `\__enumext_parse_store_keys:n` function first we will clear the variable `\l__enumext_series_str` used by the key `series` and then we check if we are at the “first level”, if so we process the `(keys)` and then execute the function `\__enumext_parse_series:n` used by the key `series` and call the function `\__enumext_nested_base_line_fix:` used by the key `base-fix`, otherwise we will pass the `(keys)` to the inner levels of the environment then we execute the function `\__enumext_store_active_keys:n` and reprocess the `(keys)` to pass them to the `sequence` if the key `save-key` is not active.

```

3659 \cs_new_protected:Npn __enumext_parse_keys:n #1
3660 {
3661 \tl_if_novalue:nF {#1}
3662 {
3663 \str_clear:N \l__enumext_series_str
3664 \int_compare:nNnTF { \l__enumext_level_int } = { 1 }
3665 {
3666 \keys_set:nn { enumext / level-1 } {#1}
3667 __enumext_parse_series:n {#1}
3668 __enumext_nested_base_line_fix:
3669 }
3670 {
3671 \exp_args:Ne \keys_set:nn
3672 { enumext / level-\int_use:N \l__enumext_level_int } {#1}
3673 }
3674 __enumext_store_active_keys:n {#1}
3675 }
3676 }

```

(End of definition for \\_\_enumext\_parse\_keys:n.)

\\_\_enumext\_start\_store\_level: The \\_\_enumext\_start\_store\_level: function activate the “*storing structure*” mechanism in the *sequence* for the command \anskey and the environment anskey\*.

```

3677 \cs_new_protected:Nn __enumext_start_store_level:
3678 {
3679 \bool_lazy_all:nT
3680 {
3681 { \bool_if_p:N \l__enumext_store_active_bool }
3682 { \bool_not_p:n { \l__enumext_keyans_env_bool } }
3683 { \bool_if_p:N \g__enumext_standar_bool }
3684 }
3685 {
3686 \int_compare:nNnT { \l__enumext_level_int } > { 1 }
3687 {
3688 \bool_set_true:c { \l__enumext_store_upper_level_ __enumext_level: _bool }
3689 __enumext_store_level_open:
3690 }
3691 }

```

If enumext are nested in enumext\* add \\_\_enumext\_store\_level\_open: to preserve the “*storing structure*”.

```

3692 \bool_lazy_all:nT
3693 {
3694 { \bool_if_p:N \l__enumext_store_active_bool }
3695 { \bool_not_p:n { \l__enumext_keyans_env_bool } }
3696 { \int_compare_p:nNn { \l__enumext_level_h_int } = { 1 } }
3697 }
3698 {
3699 \int_compare:nNnT { \l__enumext_level_int } > { 0 }
3700 {
3701 \bool_set_true:c { \l__enumext_store_upper_level_ __enumext_level: _bool }
3702 __enumext_store_level_open:
3703 }
3704 }
3705 }

```

(End of definition for \\_\_enumext\_start\_store\_level:.)

\\_\_enumext\_stop\_store\_level: The \\_\_enumext\_stop\_store\_level: function stop the “*storing structure*” mechanism in the *sequence* for the command \anskey and the environment anskey\*.

```

3706 \cs_new_protected:Nn __enumext_stop_store_level:
3707 {
3708 \bool_if:cT { \l__enumext_store_upper_level_ __enumext_level: _bool }
3709 {
3710 __enumext_store_level_close:
3711 }
3712 }

```

(End of definition for \\_\_enumext\_stop\_store\_level:.)

`\__enumext_multicols_start:` The function `\__enumext_multicols_start:` will start the `multicols` environment according to the value passed by the `columns` key, then set the default value for `\columnsep` when `columns-sep=0pt` and set the value of `\multicolsep` equal to zero and leave `\columnseprule` equal to zero for inner levels.

```

3713 \cs_new_protected:Nn __enumext_multicols_start:
3714 {
3715 \int_compare:nNt
3716 { \int_use:c { l__enumext_columns_ __enumext_level: _int } } > { 1 }
3717 {
3718 \dim_compare:nNt
3719 { \dim_use:c { l__enumext_columns_sep_ __enumext_level: _dim } } = { \c_zero_dim }
3720 {
3721 \dim_set:cn { l__enumext_columns_sep_ __enumext_level: _dim }
3722 {
3723 (\dim_use:c { l__enumext_labelwidth_ __enumext_level: _dim }
3724 + \dim_use:c { l__enumext_labelsep_ __enumext_level: _dim }
3725) / \int_use:c { l__enumext_columns_ __enumext_level: _int }
3726 - \dim_use:c { l__enumext_listoffset_ __enumext_level: _dim }
3727 }
3728 }
3729 \dim_set_eq:Nc \columnsep { l__enumext_columns_sep_ __enumext_level: _dim }
3730 \int_compare:nNt { \l__enumext_level_int } > { 1 }
3731 {
3732 \dim_zero:N \columnseprule
3733 }
3734 }

```

We will calculate the *vertical spacing* settings for the `multicols` environment using the function `\__enumext_multi_addvspace:`, apply our “*vertical adjust spacing*”, then start the `multicols` environment.

```

3734 \bool_if:cF { l__enumext_minipage_active_ __enumext_level: _bool }
3735 {
3736 \skip_zero:N \multicolsep
3737 __enumext_multi_addvspace:
3738 }
3739 \raggedcolumns
3740 \begin{multicols}{ \int_use:c { l__enumext_columns_ __enumext_level: _int } }
3741 }
3742 }

```

(End of definition for `\__enumext_multicols_start:`)

`\__enumext_multicols_stop:` The function `\__enumext_multicols_stop:` will stop the `multicols` environment and apply our “*vertical adjust*” spacing. For compatibility with *tagged* PDF, the closing of the `list` environment is executed here along with `\__enumext_stop_store_level:`.

```

3743 \cs_new_protected:Nn __enumext_multicols_stop:
3744 {
3745 \int_compare:nNtF
3746 { \int_use:c { l__enumext_columns_ __enumext_level: _int } } > { 1 }
3747 {
3748 __enumext_stop_list:
3749 __enumext_stop_store_level:
3750 \end{multicols}
3751 __enumext_unskip_unkern:
3752 __enumext_unskip_unkern:
3753 \par\addvspace{ \skip_use:c { l__enumext_multicols_below_ __enumext_level: _skip } }
3754 }
3755 {
3756 __enumext_stop_list:
3757 __enumext_stop_store_level:
3758 }
3759 }

```

(End of definition for `\__enumext_multicols_stop:`)

`\__enumext_before_list:` The function `\__enumext_before_list:` first calls the function `\__enumext_vspace_above:` used by the keys `above` and `above*`, then calls the function `\__enumext_before_args_exec:` used by the key `before*` and finally execute the function `\__enumext_check_ans_active:` for the check answer mechanism.

```

3760 \cs_new_protected:Nn __enumext_before_list:
3761 {
3762 __enumext_vspace_above:
3763 __enumext_before_args_exec:
3764 __enumext_check_ans_active:

```

When the `mini-env` key is active it will set the value of the `\l__enumext_minipage_right_X_dim` to be the *width* of the `__enumext_mini_page` environment on the “*right side*”, using this value together with the value of the `\l__enumext_minipage_hsep_X_dim` set by the `mini-sep` key, the value of `\l__enumext_minipage_left_X_dim` will be set, which will be the *width* of `__enumext_mini_page` environment on the “*left side*”, always having a current `\linewidth` as *maximum width* between them.

```

3765 \dim_compare:nNt
3766 { \dim_use:c { \l__enumext_minipage_right_ __enumext_level: _dim } } > { \c_zero_dim }
3767 {
3768 \dim_set:cn { \l__enumext_minipage_left_ __enumext_level: _dim }
3769 {
3770 \linewidth
3771 - \dim_use:c { \l__enumext_minipage_right_ __enumext_level: _dim }
3772 - \dim_use:c { \l__enumext_minipage_hsep_ __enumext_level: _dim }
3773 }

```

The boolean variable `\l__enumext_minipage_active_X_bool` will be activated and the integer variable `\g__enumext_minipage_stat_int` used by the `\miniright` command will be incremented, then the function `\__enumext_minipage_add_space:` is called and the `__enumext_mini_page` environment on the “*left side*” will be initialized followed by the “*vertical spacing*” applied to preserve the “*baseline*” between the *left* and *right* side environments. After these actions, the function `\__enumext_multicols_start:` is called to handle the `multicols` environment.

```

3774 \bool_set_true:c { \l__enumext_minipage_active_ __enumext_level: _bool }
3775 \int_gincr:N \g__enumext_minipage_stat_int
3776 __enumext_minipage_add_space:
3777 \noindent
3778 __enumext_mini_page{ \dim_use:c { \l__enumext_minipage_left_ __enumext_level: _dim } }
3779 }
3780 __enumext_multicols_start:
3781 }

```

(End of definition for `\__enumext_before_list:`)

`\__enumext_second_part:` The function `\__enumext_second_part:` first check the state of the boolean variable `\l__enumext_minipage_active_X_bool`, if it is “*true*” a small test will be executed to check if we have omitted the use of `\miniright` (the `__enumext_mini_page` environment has not been closed), then close `__enumext_mini_page` and add the *adjusted vertical space* `\l__enumext_minipage_after_skip`, otherwise we will close the `multicols` environment.

```

3782 \cs_new_protected:Nn __enumext_second_part:
3783 {
3784 \bool_if:cTF { \l__enumext_minipage_active_ __enumext_level: _bool }
3785 {
3786 \int_compare:nNt { \g__enumext_minipage_stat_int } = { 1 }
3787 {
3788 \msg_warning:nn { enumext } { missing-miniright }
3789 \miniright
3790 }
3791 \int_gzero:N \g__enumext_minipage_stat_int
3792 __enumext_unskip_unkern: % remove topsep + [partopsep]
3793 \end__enumext_mini_page
3794 }
3795 {
3796 __enumext_multicols_stop:
3797 }

```

Now we will execute the functions `\__enumext_after_stop_list:` used by the key `after`, `\__enumext_check_ans_key_hook:` used by the key `check-ans`, `\__enumext_vspace_below:` used by the keys `below` and `below*`. Finally set `\l__enumext_standar_bool` to false and call the function `\__enumext_resume_save_counter:` used by the `series`, `resume` and `resume*` keys.

```

3798 __enumext_after_stop_list:
3799 __enumext_check_ans_key_hook:
3800 __enumext_vspace_below:
3801 \bool_set_false:N \l__enumext_standar_bool
3802 __enumext_resume_save_counter:
3803 }

```

(End of definition for `\__enumext_second_part:`)

`\__enumext_set_item_width:` The function `\__enumext_set_item_width:` will set the value of `\itemwidth` taking into account the value established by the `list-offset` key for each level of the environment.

```

3804 \cs_new_protected:Nn __enumext_set_item_width:

```

```

3805 {
3806 \dim_set:Nn \itemwidth { \linewidth }
3807 \dim_compare:nT
3808 {
3809 \dim_use:c { __enumext_listoffset_ __enumext_level: _dim } != \c_zero_dim
3810 }
3811 {
3812 \dim_sub:Nn \itemwidth
3813 {
3814 \dim_use:c { __enumext_listoffset_ __enumext_level: _dim }
3815 }
3816 }
3817 }

```

(End of definition for \\_\_enumext\_set\_item\_width:.)

**enumext** Now create the **enumext** environment based on **list** environment by levels.

```

3818 \NewDocumentEnvironment{enumext}{0}{ }
3819 {
3820 __enumext_safe_exec:
3821 __enumext_parse_keys:n {#1}
3822 __enumext_before_list:
3823 __enumext_start_store_level:
3824 __enumext_start_list:nn
3825 { \tl_use:c { __enumext_label_ __enumext_level: _tl } }
3826 {
3827 \use:c { __enumext_list_arg_two_ __enumext_level: : }
3828 __enumext_before_keys_exec:
3829 }
3830 __enumext_set_item_width:
3831 __enumext_after_args_exec:
3832 }
3833 {
3834 __enumext_second_part:
3835 }

```

(End of definition for enumext. This function is documented on page 5.)

As we don't want our check to be executed **check-ans** by levels but on the complete list, we will take it out of the **enumext** environment using the “hook” function \\_\_enumext\_after\_env:nn.

```

3836 __enumext_after_env:nn {enumext}
3837 {
3838 __enumext_execute_after_env:
3839 }

```

### 13.40 The environment keyans

The environment **keyans** also based on lists. The main differences with the **enumext** environment are the *nesting* and the way the *answers* (choice) will be stored and checked, this environment is intended exclusively for “multiple choice questions”.

**\\_\_enumext\_keyans\_safe\_exec:** The **keyans** environment will only be available if the **save-ans** key is active and can only be used at the “first level” within the **enumext** environment. We do not want the environment to be nested, so we will set a maximum at this point. If the conditions are not met, an error message will be returned.

```

3840 \cs_new_protected:Nn __enumext_keyans_safe_exec:
3841 {
3842 \bool_if:NF \l__enumext_store_active_bool
3843 {
3844 \msg_error:nnnn { enumext } { wrong-place } { keyans } { save-ans }
3845 }
3846 \int_incr:N \l__enumext_keyans_level_int
3847 \bool_set_true:N \l__enumext_keyans_env_bool
3848 __enumext_keyans_name_and_start:
3849 % Set false for interfering with enumext nested in keyans (yes, its possible and crayze)
3850 \bool_set_false:N \l__enumext_store_active_bool
3851 \int_compare:nNnT { \l__enumext_keyans_level_int } > { 1 }
3852 {
3853 \msg_error:nn { enumext } { keyans-nested }
3854 }
3855 \int_compare:nNnT { \l__enumext_level_int } > { 1 }
3856 {
3857 \msg_error:nn { enumext } { keyans-wrong-level }

```

```

3858 }
3859 }

```

(End of definition for `\__enumext_keyans_safe_exec:`)

```
__enumext_keyans_parse_keys:n
```

Parse [*key* = *val*] for `keyans` environment.

```

3860 \cs_new_protected:Npn __enumext_keyans_parse_keys:n #1
3861 {
3862 \keys_set:nn { enumext / keyans } {#1}
3863 }

```

(End of definition for `\__enumext_keyans_parse_keys:n`)

```
__enumext_before_list_v:
```

Same implementation as the one used in the `enumext` environment.

```
__enumext_keyans_multicols_start:
```

```
3864 \cs_new_protected:Nn __enumext_before_list_v:
```

```
__enumext_keyans_multicols_stop:
```

```
3865 {
```

```
__enumext_second_part_v:
```

```
3866 __enumext_vspace_above_v:
```

```
3867 __enumext_before_args_exec_v:
```

```
3868 \dim_compare:nNtT { \l__enumext_minipage_right_v_dim } > { \c_zero_dim }
```

```
3869 {
```

```
3870 \dim_set:Nn \l__enumext_minipage_left_v_dim
```

```
3871 {
```

```
3872 \linewidth - \l__enumext_minipage_right_v_dim - \l__enumext_minipage_hsep_v_dim
```

```
3873 }
```

```
3874 \bool_set_true:N \l__enumext_minipage_active_v_bool
```

```
3875 \int_gincr:N \g__enumext_minipage_stat_int
```

```
3876 __enumext_keyans_minipage_add_space:
```

```
3877 __enumext_mini_page{ \l__enumext_minipage_left_v_dim }
```

```
3878 }
```

```
3879 __enumext_keyans_multicols_start:
```

```
3880 }
```

```
3881 \cs_new_protected:Nn __enumext_keyans_multicols_start:
```

```
3882 {
```

```
3883 \int_compare:nNtT { \l__enumext_columns_v_int } > { 1 }
```

```
3884 {
```

```
3885 \dim_compare:nNtT { \l__enumext_columns_sep_v_dim } = { \c_zero_dim }
```

```
3886 {
```

```
3887 \dim_set:Nn \l__enumext_columns_sep_v_dim
```

```
3888 {
```

```
3889 (
```

```
3890 \l__enumext_labelwidth_v_dim + \l__enumext_labelsep_v_dim
```

```
3891) / \l__enumext_columns_v_int
```

```
3892 - \l__enumext_listoffset_v_dim
```

```
3893 }
```

```
3894 }
```

```
3895 \dim_set_eq:NN \columnsep \l__enumext_columns_sep_v_dim
```

```
3896 \dim_zero:N \columnseprule % no rule here
```

```
3897 \bool_if:NF \l__enumext_minipage_active_v_bool
```

```
3898 {
```

```
3899 \skip_zero:N \multicolsep
```

```
3900 __enumext_keyans_multi_addvspace:
```

```
3901 }
```

```
3902 \raggedcolumns
```

```
3903 \begin{multicols}{\l__enumext_columns_v_int }
```

```
3904 }
```

```
3905 }
```

```
3906 \cs_new_protected:Nn __enumext_keyans_multicols_stop:
```

```
3907 {
```

```
3908 \int_compare:nNtTF { \l__enumext_columns_v_int } > { 1 }
```

```
3909 {
```

```
3910 __enumext_stop_list:
```

```
3911 \end{multicols}
```

```
3912 __enumext_unskip_unkern:
```

```
3913 __enumext_unskip_unkern:
```

```
3914 \par\addvspace{ \l__enumext_multicols_below_v_skip }
```

```
3915 }
```

```
3916 {
```

```
3917 __enumext_stop_list:
```

```
3918 }
```

```
3919 }
```

```
3920 \cs_new_protected:Nn __enumext_second_part_v:
```

```

3921 {
3922 \bool_if:NTF \l__enumext_minipage_active_v_bool
3923 {
3924 \int_compare:nNt { \g__enumext_minipage_stat_int } = { 1 }
3925 {
3926 \msg_warning:nn { enumext } { missing-miniright }
3927 \miniright
3928 }
3929 \int_gzero:N \g__enumext_minipage_stat_int
3930 __enumext_unskip_unkern: % remove \topsep + [\partopsep]
3931 \end__enumext_mini_page
3932 \par\addvspace{ \l__enumext_minipage_after_skip }
3933 }
3934 {
3935 __enumext_keyans_multicols_stop:
3936 }
3937 \bool_set_false:N \l__enumext_keyans_env_bool
3938 __enumext_after_stop_list_v:
3939 __enumext_vspace_below_v:
3940 }

```

(End of definition for \\_\_enumext\_before\_list\_v: and others.)

\\_\_enumext\_keyans\_set\_item\_width: The function \\_\_enumext\_keyans\_set\_item\_width: will set the value of \itemwidth taking into account the value established by the list-offset key.

```

3941 \cs_new_protected:Nn __enumext_keyans_set_item_width:
3942 {
3943 \dim_set:Nn \itemwidth { \linewidth }
3944 \dim_compare:nT
3945 {
3946 \l__enumext_listoffset_v_dim != \c_zero_dim
3947 }
3948 {
3949 \dim_sub:Nn \itemwidth { \l__enumext_listoffset_v_dim }
3950 }
3951 }

```

(End of definition for \\_\_enumext\_keyans\_set\_item\_width:.)

**keyans** Now we define the environment **keyans** also based on lists.

```

3952 \NewDocumentEnvironment{keyans}{0}{}
3953 {
3954 __enumext_keyans_safe_exec:
3955 __enumext_keyans_parse_keys:n {#1}
3956 __enumext_before_list_v:
3957 __enumext_start_list:nn
3958 { \tl_use:N \l__enumext_label_v_tl }
3959 {
3960 __enumext_list_arg_two_v:
3961 __enumext_before_keys_exec_v:
3962 }
3963 __enumext_keyans_set_item_width:
3964 __enumext_after_args_exec_v:
3965 }
3966 {
3967 __enumext_check_starred_cmd:n { item }
3968 __enumext_second_part_v:
3969 }

```

(End of definition for keyans. This function is documented on page 14.)

### 13.41 Tagging PDF support for non-standart list environments

The  $\TeX$  release 2022-06-01 brings automatic support for *tagged* PDF in several aspects, including the standard *list environments* and the **list** environment. Unfortunately non-standard *list environments* like **keyanspic** or the horizontal list environments **enumext\*** and **keyans\*** are not structured in a nice way, i.e. the expected result in the PDF file is the expected one, but the underlying structure is not correct. In simple terms, for *tagged* PDF a **list** environment is a **list** environment, no matter what it looks like in the PDF file.

To maintain a correct **list** structure when \DocumentMetadata is active, it is necessary to do some things manually using **tagpdf**[17] and **ltsockets**[19]. This implementation is an adaptation of my answer thanks to Ulrike Fischer's comments in [How can I modify my \item redefinition to be compatible with tagging-pdf](#).



### 13.41.1 Socket for tagging support in enumext\* and keyans\*

```
start-list-tags
stop-start-tags
stop-list-tags
__enumext_start_list_tag:n
 __enumext_stop_start_list_tag:
 __enumext_stop_list_tag:n
```

We will first define the necessary sockets and their behavior for `enumext*` and `keyans*`.

```
3970 \socket_new:nn {tagsupport/__enumext/starred}{ 1 }
3971 \socket_new_plugin:nnn {tagsupport/__enumext/starred} {start-list-tags}
3972 {
3973 \tag_resume:n {#1}
3974 \tag_mc_end_push:
3975 \tag_struct_begin:n {tag=LI}
3976 \tag_struct_begin:n {tag=Lbl}
3977 \tag_mc_begin:n {tag=Lbl}
3978 }
3979 \socket_new_plugin:nnn {tagsupport/__enumext/starred} {stop-start-tags}
3980 {
3981 \tag_mc_end:
3982 \tag_struct_end:n {tag=Lbl}
3983 \tag_struct_begin:n {tag=LBody}
3984 \tag_struct_begin:n {tag=text-unit}
3985 \tag_struct_begin:n {tag=text}
3986 }
3987 \socket_new_plugin:nnn {tagsupport/__enumext/starred} {stop-list-tags}
3988 {
3989 \tag_struct_end:n {tag=text}
3990 \tag_struct_end:n {tag=text-unit}
3991 \tag_struct_end:n {tag=LBody}
3992 \tag_struct_end:n {tag=LI}
3993 \tag_mc_begin_pop:n {}
3994 \tag_suspend:n {#1}
3995 }
```

And now we'll wrap them so that they're only active when `\DocumentMetadata` is present.

```
3996 \cs_new_protected_nopar:Npn __enumext_start_list_tag:n #1
3997 {
3998 \IfDocumentMetadataTF
3999 {
4000 \socket_assign_plugin:nn {tagsupport/__enumext/starred} {start-list-tags}
4001 \socket_use:nn {tagsupport/__enumext/starred} {#1}
4002 } {}
4003 }
4004 \cs_new_protected_nopar:Npn __enumext_stop_start_list_tag:
4005 {
4006 \IfDocumentMetadataTF
4007 {
4008 \socket_assign_plugin:nn {tagsupport/__enumext/starred} {stop-start-tags}
4009 \socket_use:nn {tagsupport/__enumext/starred} {}
4010 } {}
4011 }
4012 \cs_new_protected_nopar:Npn __enumext_stop_list_tag:n #1
4013 {
4014 \IfDocumentMetadataTF
4015 {
4016 \socket_assign_plugin:nn {tagsupport/__enumext/starred} {stop-list-tags}
4017 \socket_use:nn {tagsupport/__enumext/starred} {#1}
4018 } {}
4019 }
```

*(End of definition for start-list-tags and others.)*

### 13.41.2 Socket for tagging support in keyanspic

```
start-list-tags
stop-start-tags
stop-list-tags
 __enumext_anspic_start_list_tag:
 __enumext_anspic_stop_start_list_tag:
 __enumext_anspic_stop_list_tag:
```

We will first define the necessary sockets and their behavior for `keyanspic` environment.

```
4020 \socket_new:nn {tagsupport/__enumext/keyanspic}{ 0 }
4021 \socket_new_plugin:nnn {tagsupport/__enumext/keyanspic} {start-list-tags}
4022 {
4023 \tag_resume:n {keyanspic}
4024 \tag_mc_end_push:
4025 \tag_struct_begin:n {tag=LI}
4026 \tag_struct_begin:n {tag=Lbl}
4027 \tag_mc_begin:n {tag=Lbl}
4028 }
4029 \socket_new_plugin:nnn {tagsupport/__enumext/keyanspic} {stop-start-tags}
4030 {
4031 \tag_mc_end:
```

```
4032 \tag_struct_end:n {tag=Lbl}
4033 \tag_struct_begin:n {tag=LBody}
4034 \tag_struct_begin:n {tag=text-unit}
4035 \tag_struct_begin:n {tag=text}
4036 \tag_mc_begin:n {tag=text}
4037 }
4038 \socket_new_plug:nnn {tagsupport/__enumext/keyanspic} {stop-list-tags}
4039 {
4040 \tag_mc_end:
4041 \tag_struct_end:n {tag=text}
4042 \tag_struct_end:n {tag=text-unit}
4043 \tag_struct_end:n {tag=LBody}
4044 \tag_struct_end:n {tag=LI}
4045 \tag_mc_begin_pop:n {}
4046 \tag_suspend:n {keyanspic}
4047 }
```

And now we'll wrap them so that they're only active when \DocumentMetadata is present.

```
4048 \cs_new_protected_nopar:Nn __enumext_anspic_start_list_tag:
4049 {
4050 \IfDocumentMetadataTF
4051 {
4052 \socket_assign_plug:nn {tagsupport/__enumext/keyanspic} {start-list-tags}
4053 \socket_use:n {tagsupport/__enumext/keyanspic}
4054 } {}
4055 }
4056 \cs_new_protected_nopar:Nn __enumext_anspic_stop_start_list_tag:
4057 {
4058 \IfDocumentMetadataTF
4059 {
4060 \socket_assign_plug:nn {tagsupport/__enumext/keyanspic} {stop-start-tags}
4061 \socket_use:n {tagsupport/__enumext/keyanspic}
4062 } {}
4063 }
4064 \cs_new_protected_nopar:Nn __enumext_anspic_stop_list_tag:
4065 {
4066 \IfDocumentMetadataTF
4067 {
4068 \socket_assign_plug:nn {tagsupport/__enumext/keyanspic} {stop-list-tags}
4069 \socket_use:n {tagsupport/__enumext/keyanspic}
4070 } {}
4071 }
```

(End of definition for start-list-tags and others.)

13.42 The environment keyanspic and \anspic

The `keyanspic` environment is a `list` based environment that uses the same configuration for “spacing” and `<label>` as the `keyans` environment, but it does not use `\item`. The `<contents>` are passed to the environment by means of the `\anspic` command as replacement for `\item` command and placed inside `minipage` environments, with the `<label>` centered “above” or “below”, adjusting *widths* and *position* according to the options passed to the environment.

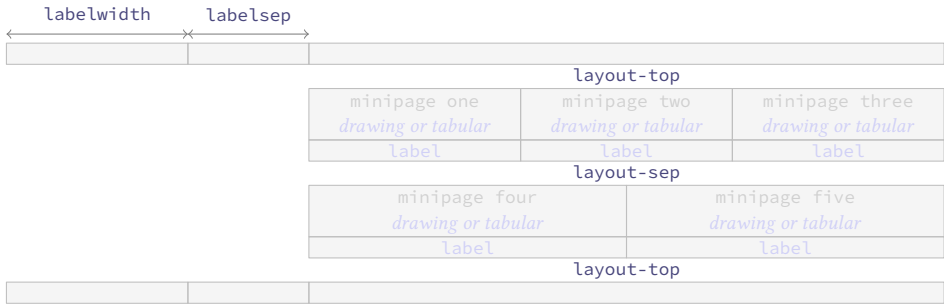


Figure 12: Representation of the `keyanspic` spacing in `enumext`.

In order for the `keyanspic` environment and the `\anspic` command to work correctly, we need to set and export some variables in the first part of the environment definition and pass them to `\anspic` which is executed in the second part of the environment. This implementation is adapted from the answer given by Enrico Gregorio (@egreg) in [How to process the body of an environment and divide it by a \macro?](#).

13.42.1 The environment keyanspic

First we define the key that allows us to process the position of the `<label>` centered “above” or “below” which will be `label-pos`, the vertical separation of these from *drawing or tabular* will be handled with the key

**label-sep.** The “*layout style*” will be handled with the key `layout-sty` will take two values separated by comma  $\{ \langle n^{\circ} \textit{upper}, n^{\circ} \textit{lower} \rangle \}$  and will determine the number of `minipage` environments in which all arguments of `\anspic` will be printed at the “upper” and “lower” within the environments separated by the value of the key `layout-sep`. The vertical space “top” and “bottom” of the environment will be handled with the key `layout-top`.

```

4072 \keys_define:nn { enumext / keyanspic }
4073 {
4074 label-pos .choice:,
4075 label-pos / above .code:n =
4076 \bool_set_true:N \l__enumext_anspic_label_above_bool
4077 \str_set:Nn \l__enumext_anspic_mini_pos_str { t },
4078 label-pos / below .code:n =
4079 \bool_set_false:N \l__enumext_anspic_label_above_bool
4080 \str_set:Nn \l__enumext_anspic_mini_pos_str { b },
4081 label-pos / unknown .code:n =
4082 \msg_error:nneee { enumext } { unknown-choice }
4083 { label-pos } { above,~ below } { \exp_not:n {#1} },
4084 label-pos .initial:n = below,
4085 label-pos .value_required:n = true,
4086 label-sep .skip_set:N = \l__enumext_anspic_label_sep_skip,
4087 label-sep .value_required:n = true,
4088 layout-sty .tl_set:N = \l__enumext_anspic_layout_style_tl,
4089 layout-sty .value_required:n = true,
4090 layout-sep .code:n = \keys_set:nn { enumext / keyans }
4091 { parsep = #1 },
4092 layout-sep .value_required:n = true,
4093 layout-top .code:n = \keys_set:nn { enumext / keyans }
4094 { topsep = #1 },
4095 layout-top .value_required:n = true,
4096 unknown .code:n = { \l__enumext_keyans_unknown_keys:n {#1} }
4097 }

```

(End of definition for `label-pos` and others.)

```

__enumext_keyans_pic_safe_exec:
__enumext_keyans_pic_parse_keys:n
__enumext_keyans_pic_skip_abs:N
__enumext_keyans_pic_arg_two:

```

The function `\__enumext_keyans_pic_safe_exec:` check the nested level position inside the `enumext` environment.

```

4098 \cs_new_protected:Nn __enumext_keyans_pic_safe_exec:
4099 {
4100 \int_incr:N \l__enumext_keyans_pic_level_int
4101 \int_compare:nNnT { \l__enumext_keyans_pic_level_int } > { 1 }
4102 {
4103 \msg_error:nn { enumext } { keyanspic-nested }
4104 }
4105 __enumext_keyans_name_and_start:
4106 }

```

Parse  $[\langle \textit{key} = \textit{val} \rangle]$  for `keyanspic` environment.

```

4107 \cs_new_protected:Npn __enumext_keyans_pic_parse_keys:n #1
4108 {
4109 \tl_if_novalue:nF {#1}
4110 {
4111 \keys_set:nn { enumext / keyanspic } {#1}
4112 }
4113 }

```

The function `\__enumext_keyans_pic_skip_abs:N` will return a positive value `\parsep` from `keyans` environment.

```

4114 \cs_new_protected:Npn __enumext_keyans_pic_skip_abs:N #1
4115 {
4116 \dim_compare:nNnT { #1 } < { \c_zero_dim }
4117 {
4118 \skip_set:Nn #1 { -#1 }
4119 }
4120 }

```

The `\__enumext_keyans_pic_arg_two:` function will be used in the *second argument* of the `list` environment that defines the `keyanspic` environment, with this we will take the configuration of the “*spaces*” and the keys `label`, `wrap-label`, `parsep` and `topsep` from the `keyans` environment. The first thing we need to do is set the boolean variable `\l__enumext_leftmargin_tmp_v_bool` handled by the `list-indent` key to “false”, then copy the definition of the second list argument from the `keyans` environment definition and make sure that `\parsep` does not have a negative value.

```

4121 \cs_new_protected:Npn __enumext_keyans_pic_arg_two:
4122 {
4123 \bool_set_false:N \l__enumext_leftmargin_tmp_v_bool
4124 __enumext_list_arg_two_v:
4125 __enumext_keyans_pic_skip_abs:N \parsep

```

Now we increment the counter `enumXv` of the `keyans` environment and save the *total height* of the *(label)* in `\l__enumext_anspic_label_htdp_dim` used by `\anspic` and we will adjust the values of `\parsep` only if the key `label-pos` is set to *below*.

```

4126 \bool_if:NF \l__enumext_anspic_label_above_bool
4127 {
4128 \stepcounter { enumXv }
4129 \hbox_set:Nn \l__enumext_anspic_label_box { \l__enumext_label_v_tl }
4130 \dim_set:Nn \l__enumext_anspic_label_htdp_dim
4131 {
4132 \box_ht_plus_dp:N \l__enumext_anspic_label_box
4133 }
4134 \skip_add:Nn \parsep
4135 {
4136 \l__enumext_anspic_label_htdp_dim
4137 + \box_dp:N \strutbox
4138 + \l__enumext_anspic_label_sep_skip
4139 }
4140 }

```

Finally we *adjust* the value of `\leftmargin` and `\topsep` then set `\listparindent`, `\partopsep` and `\itemsep` to zero so that the *horizontal* and *vertical* space is not affected.

```

4141 \dim_add:Nn \leftmargin { -\labelwidth - \labelsep }
4142 \skip_add:Nn \topsep { 0.5\box_dp:N \strutbox }
4143 \dim_zero:N \listparindent
4144 \skip_zero:N \partopsep
4145 \skip_zero:N \itemsep
4146 }

```

(End of definition for `\__enumext_keyans_pic_safe_exec:` and others.)

**keyanspic** Now we define the environment `keyanspic`. For compatibility with *tagged* PDF we must use the `\begin{list}` form and a lot of conditional code using `\IfDocumentMetadataTF`. We will first stop the code for automatic *tagged* PDF for `list` environments, redefine `\item` so that it cannot be used, and stop the code for automatic *tagged* PDF for the `keyanspic` environment.

```

4147 \NewDocumentEnvironment{keyanspic}{ o }
4148 {
4149 __enumext_keyans_pic_safe_exec:
4150 __enumext_keyans_pic_parse_keys:n {#1}
4151 \begin{list} { } { __enumext_keyans_pic_arg_two: }
4152 \IfDocumentMetadataTF
4153 {
4154 \tag_suspend:n {list}
4155 }{}
4156 \item[] \scan_stop:
4157 \RenewDocumentCommand \item {}
4158 {
4159 \msg_error:nn { enumext } { keyanspic-item-cmd }
4160 }
4161 \IfDocumentMetadataTF
4162 {
4163 \tag_resume:n {keyanspic}
4164 \tag_tool:n {para/tagging=false}
4165 \tag_suspend:n {keyanspic}
4166 } { }
4167 }

```

In the second part of the environment definition we will manually place our code for *tagged* PDF and execute the command `\anspic` using the `\__enumext_anspic_exec:` function.

```

4168 {
4169 \IfDocumentMetadataTF
4170 {
4171 \tag_resume:n {keyanspic}
4172 \tag_mc_end_push:
4173 \tag_struct_begin:n {tag=L,attribute=enumerate}
4174 } { }
4175 __enumext_anspic_exec:

```

```

4176 \IfDocumentMetadataTF
4177 {
4178 \tag_suspend:n {keyanspic}
4179 } { }
4180 \end{list}
4181 \IfDocumentMetadataTF
4182 {
4183 \tag_struct_end:n {tag=L}
4184 \tag_mc_begin_pop:n {}
4185 \tag_struct_end:n {tag=L}
4186 \tag_mc_begin_pop:n {}
4187 } { }

```

Finally we check if `\anspic*` has been used, set the counter `enumXvi` to zero and apply our “adjusted” vertical space bottom.

```

4188 __enumext_check_starred_cmd:n { anspic }
4189 \setcounter { enumXvi } { 0 }
4190 \bool_if:NTF \l__enumext_anspic_label_above_bool
4191 {
4192 \par\addvspace{ 0.5\box_dp:N \strutbox }
4193 }
4194 {
4195 \par
4196 \addvspace
4197 {
4198 \dim_eval:n
4199 {
4200 \l__enumext_anspic_label_htdp_dim + \box_ht_plus_dp:N \strutbox
4201 + \l__enumext_anspic_label_sep_skip + \l__enumext_topsep_v_skip
4202 }
4203 }
4204 }
4205 }

```

(End of definition for `keyanspic`. This function is documented on page 15.)

### 13.42.2 The command `\anspic`

The `\anspic` command take three arguments, the *starred versions* `\anspic*[\langle content \rangle]` store the current `\label` next to the *optional argument* `[\langle content \rangle]` in the *sequence* and *prop list* defined by `save-ans` key. The third *mandatory argument* `{\langle drawing or tabular \rangle}` is NOT stored in the *sequence* or *prop list*.

- One of the complications here to make the `keyanspic` environment compatible with *tagged* PDF is the position of `\label`, the `\anspic` command processes the arguments in order, where #1 and #2 correspond to `\label` and #3 to the mandatory argument and puts all this inside a `minipage` environment. If #1 and #2, that is `\label`, is above #3 there are no problems with *tagged* PDF, but if #3 comes first the list created with *tagged* PDF will not be correct.

`\anspic`

We check that the command is active in the `keyanspic` environment only if the `save-ans` key is present, otherwise we return an error. The three arguments are handled by the function `\__enumext_anspic_args:nnn` and stored in the sequence `\l__enumext_anspic_args_seq` which is processed by the `keyanspic` environment.

```

4206 \NewDocumentCommand \anspic { s o +m }
4207 {
4208 \bool_if:NF \l__enumext_store_active_bool
4209 {
4210 \msg_error:nnnn { enumext } { wrong-place } { keyanspic } { save-ans }
4211 }
4212 \int_compare:nNt { \l__enumext_level_int } > { 1 }
4213 {
4214 \msg_error:nn { enumext } { keyanspic-wrong-level }
4215 }
4216 \int_compare:nNt { \l__enumext_keyans_level_int } = { 1 }
4217 {
4218 \msg_error:nnnn { enumext } { command-wrong-place } { anspic } { keyans }
4219 }
4220 \seq_put_right:Nn \l__enumext_anspic_args_seq
4221 {
4222 __enumext_anspic_args:nnn { #1 } { #2 } { #3 }
4223 }
4224 }

```

The `\__enumext_anspic_body_dim:n` function will set the value of `\l__enumext_anspic_body_htdp_dim` equal to the “height plus depth” of the *mandatory argument* if the key `label-pos` is set “below”.

```

4225 \cs_new_protected:Npn __enumext_anspic_body_dim:n #1
4226 {
4227 \bool_if:NF \l__enumext_anspic_label_above_bool
4228 {
4229 \IfDocumentMetadataTF
4230 {
4231 \tag_suspend:n {keyanspic}
4232 } { }
4233 \vbox_set:Nn \l__enumext_anspic_body_box { #1 }
4234 \dim_set:Nn \l__enumext_anspic_body_htdp_dim
4235 {
4236 \box_ht_plus_dp:N \l__enumext_anspic_body_box
4237 }
4238 \IfDocumentMetadataTF
4239 {
4240 \tag_resume:n {keyanspic}
4241 } { }
4242 }
4243 }

```

The `\__enumext_anspic_label:nn` function will process inside `\makebox` the *starred argument* ‘`*`’ and *optional argument* passed to the command. Here we will store the `<label>` and *optional argument* in *prop list* and *sequence* and execute the `show-ans`, `show-pos`, `font`, `wrap-label`, `wrap-key` and `wrap-opt` keys.

```

4244 \cs_new_protected:Npn __enumext_anspic_label:nn #1 #2
4245 {
4246 \makebox[\l__enumext_anspic_mini_width_dim][c]
4247 {
4248 \bool_if:nTF { #1 }
4249 {
4250 \int_incr:N \l__enumext_item_star_exec_int
4251 \bool_set_true:N \l__enumext_wrap_label_v_bool
4252 __enumext_keyans_addto_prop:n { #2 }
4253 __enumext_keyans_store_ref:
4254 __enumext_keyans_addto_seq:n { #2 }
4255 \int_gincr:N \g__enumext_check_starred_cmd_int
4256 \bool_lazy_or:nnT
4257 { \bool_if_p:N \l__enumext_show_answer_bool }
4258 { \bool_if_p:N \l__enumext_show_position_bool }
4259 {
4260 \tl_set_eq:NN \l__enumext_label_v_tl \l__enumext_label_vi_tl
4261 __enumext_keyans_show_left:n { #2 }
4262 \tl_set_eq:NN \l__enumext_label_vi_tl \l__enumext_label_v_tl
4263 }
4264 \tl_use:N \l__enumext_label_font_style_v_tl
4265 __enumext_keyans_wrapper_label:n { \l__enumext_label_vi_tl }
4266 \c_space_tl __enumext_keyans_show_item_opt:
4267 }
4268 {
4269 \int_zero:N \l__enumext_item_star_exec_int % zero
4270 \tl_use:N \l__enumext_label_font_style_v_tl
4271 __enumext_wrapper_label_v:n { \l__enumext_label_vi_tl }
4272 }
4273 }
4274 }

```

The function `\__enumext_anspic_label_pos:nnn` will be in charge of handling the “*counter*” and the position of the `<label>`, set by `label-pos` key which will have the same configuration as the `keyans` environment.

```

4275 \cs_new_protected:Npn __enumext_anspic_label_pos:nnn #1 #2 #3
4276 {
4277 \stepcounter { enumXvi }
4278 __enumext_anspic_body_dim:n { #3 }
4279 \bool_if:NTF \l__enumext_anspic_label_above_bool
4280 {
4281 __enumext_anspic_label:nn { #1 } { #2 }
4282 }
4283 {
4284 \raisebox
4285 {
4286 -\dim_eval:n
4287 {
4288 \l__enumext_anspic_label_htdp_dim
4289 + \l__enumext_anspic_body_htdp_dim

```

```

4290 + \box_dp:N \strutbox
4291 + \l__enumext_anspic_label_sep_skip
4292 }
4293 }
4294 [opt] [opt]
4295 {
4296 __enumext_anspic_label:nn { #1 } { #2 }
4297 }
4298 }
4299 }
4300 %

```

The `\__enumext_anspic_args:nnn` function will be responsible for placing the code compatible with *tagged* PDF and the arguments within the `\l__enumext_anspic_args_seq` sequence which will be processed by the `\__enumext_anspic_print:n` function in the second part of the definition of the `keyanspic` environment.

```

4301 \cs_new_protected:Nn __enumext_anspic_args:nnn
4302 {
4303 __enumext_anspic_start_list_tag:
4304 __enumext_anspic_label_pos:nnn { #1 } { #2 } { #3 }
4305 __enumext_anspic_stop_start_list_tag:
4306 \bool_if:NTF \l__enumext_anspic_label_above_bool
4307 {
4308 \[\l__enumext_anspic_label_sep_skip] #3
4309 }
4310 {
4311 \[#3
4312 }
4313 __enumext_anspic_stop_list_tag:
4314 }

```

The value  $\{ \langle n^{\circ upper}, n^{\circ lower} \rangle \}$  passed to the `layout-sty` key is split by comma and is handled directly by the function `\__enumext_anspic_print:n` and passed to the function `\__enumext_anspic_row:n`.

```

4315 \cs_new_protected:Nn __enumext_anspic_print:n
4316 {
4317 \clist_map_function:nN { #1 } __enumext_anspic_row:n
4318 }
4319 \cs_generate_variant:Nn __enumext_anspic_print:n { e, V }

```

The function `\__enumext_anspic_row:n` will set the *widths* for the `minipage` environments and place *all arguments* passed to `\anspic saved` in the `\l__enumext_anspic_args_seq` sequence inside them.

```

4320 \cs_new_protected:Nn __enumext_anspic_row:n
4321 {
4322 \dim_set:Nn \l__enumext_anspic_mini_width_dim { \linewidth / #1 }
4323 \int_set:Nn \l__enumext_anspic_above_int { \l__enumext_anspic_below_int }
4324 \int_set:Nn \l__enumext_anspic_below_int { \l__enumext_anspic_above_int + #1 }
4325 \int_step_inline:nnn
4326 { \l__enumext_anspic_above_int + 1 }
4327 { \l__enumext_anspic_below_int }
4328 {
4329 \IfDocumentMetadataTF
4330 {
4331 \tag_suspend:n {minipage}
4332 } { }
4333 \begin{minipage}[\l__enumext_anspic_mini_pos_str]{ \l__enumext_anspic_mini_width_dim }
4334 \centering
4335 \seq_item:Nn \l__enumext_anspic_args_seq { ##1 }
4336 \end{minipage}
4337 \IfDocumentMetadataTF
4338 {
4339 \tag_resume:n {minipage}
4340 } { }
4341 }
4342 \par
4343 }

```

The `\__enumext_anspic_exec:` function will execute all the code in the `\anspic` command in the second argument of the `keyanspic` environment definition. If the key `layout-sty` is not set, everything will be printed on a *single line*.

```

4344 \cs_new_protected:Nn __enumext_anspic_exec:
4345 {
4346 \tl_if_empty:NTF \l__enumext_anspic_layout_style_tl
4347 {

```



```

4348 __enumext_anspic_print:e { \seq_count:N __enumext_anspic_args_seq }
4349 }
4350 {
4351 __enumext_anspic_print:V __enumext_anspic_layout_style_tl
4352 }
4353 }

```

(End of definition for `\anspic` and others. This function is documented on page 16.)

### 13.43 The horizontal environments

Generating *horizontal list environments* is NOT as simple as standard  $\text{\LaTeX}$  list environments. The fundamental part of the code is adapted from the `shortlist` package to a more modern version using `expl3`. It is not possible to redefine `\item` and `\makelabel` using `\RenewDocumentCommand` as in the vertical *non starred* versions.

To achieve the *horizontal list environments* we will capture the `\item` command and the  $\langle content \rangle$  of this in *horizontal box* using `\makebox` for the *label* and a `minipage` environment for the  $\langle content \rangle$  passed to `\item`, we will also add the *optional argument*  $\langle number \rangle$  to `\item` to be able to *join columns* horizontally, in simple terms, we want `\item` to behave in the same way as in the `enumext` environment but adding an *first optional argument*  $\langle number \rangle$ .

A side effect is the limitation of using `\item` in this way *without* using `\RenewDocumentCommand`, which loses the original definition and affects the *standard list environments* provided by  $\text{\LaTeX}$  and any environment defined using base `list` environment, including: `itemize`, `enumerate`, `description`, `quote`, `quotation`, `verse`, `center`, `flushleft`, `flushright`, `verbatim`, `tabbing`, `trivlist`, `list` and all environments created with `\newtheorem`.

One way to get around this is to use something like:

```
\AddToHook{env/enumerate/before}{recover original \item definition}
```

inside `minipage`, but in my partial tests this does not have the desired effect and the vertical and horizontal spacing is distorted. For now this will remain as a limitation and I will see if it is feasible to implement it in the future.

For compatibility with the *tagged* PDF we close the environments according to the presence or not of the `mini-env` key.

#### 13.43.1 Functions for item box width

We set the default value for the *width of the box* containing the  $\langle content \rangle$  of the items for `enumext*` environment.

```

__enumext_starred_columns_set_vii:
__enumext_starred_columns_set_viii:
4354 \cs_new_protected:Nn __enumext_starred_columns_set_vii:
4355 {
4356 \dim_compare:nNnT { \l__enumext_columns_sep_vii_dim } = { \c_zero_dim }
4357 {
4358 \dim_set:Nn \l__enumext_columns_sep_vii_dim
4359 {
4360 (\l__enumext_labelwidth_vii_dim + \l__enumext_labelsep_vii_dim)
4361 / \l__enumext_columns_vii_int
4362 }
4363 }
4364 \int_set:Nn \l__enumext_tmpa_vii_int { \l__enumext_columns_vii_int - 1 }
4365 \dim_set:Nn \l__enumext_item_width_vii_dim
4366 {
4367 (\linewidth - \l__enumext_columns_sep_vii_dim * \l__enumext_tmpa_vii_int)
4368 / \l__enumext_columns_vii_int
4369 - \l__enumext_labelwidth_vii_dim
4370 - \l__enumext_labelsep_vii_dim
4371 }

```

When the key `rightmargin` is active we must adjust the values.

```

4372 \dim_compare:nNnT { \l__enumext_rightmargin_vii_dim } > { \c_zero_dim }
4373 {
4374 \dim_sub:Nn \l__enumext_item_width_vii_dim
4375 {
4376 (\l__enumext_rightmargin_vii_dim * \l__enumext_tmpa_vii_int)
4377 / \l__enumext_columns_vii_int
4378 }
4379 \dim_add:Nn \l__enumext_columns_sep_vii_dim
4380 {
4381 \l__enumext_rightmargin_vii_dim
4382 }
4383 }
4384 }

```

Same implementation for the `keyans*` environment.

```

4385 \cs_new_protected:Nn __enumext_starred_columns_set_viii:
4386 {
4387 \dim_compare:nNnT { __enumext_columns_sep_viii_dim } = { \c_zero_dim }
4388 {
4389 \dim_set:Nn __enumext_columns_sep_viii_dim
4390 {
4391 (__enumext_labelwidth_viii_dim + __enumext_labelsep_viii_dim)
4392 / __enumext_columns_viii_int
4393 }
4394 }
4395 \int_set:Nn __enumext_tmpa_viii_int { __enumext_columns_viii_int - 1 }
4396 \dim_set:Nn __enumext_item_width_viii_dim
4397 {
4398 (\linewidth - __enumext_columns_sep_viii_dim * __enumext_tmpa_viii_int)
4399 / __enumext_columns_viii_int
4400 - __enumext_labelwidth_viii_dim
4401 - __enumext_labelsep_viii_dim
4402 }
4403 \dim_compare:nNnT { __enumext_rightmargin_viii_dim } > { \c_zero_dim }
4404 {
4405 \dim_sub:Nn __enumext_item_width_viii_dim
4406 {
4407 (__enumext_rightmargin_viii_dim * __enumext_tmpa_viii_int)
4408 / __enumext_columns_viii_int
4409 }
4410 \dim_add:Nn __enumext_columns_sep_viii_dim
4411 {
4412 __enumext_rightmargin_viii_dim
4413 }
4414 }
4415 }

```

(End of definition for `\__enumext_starred_columns_set_vii:` and `\__enumext_starred_columns_set_viii:`)

### 13.43.2 Functions for join item columns

`\__enumext_starred_joined_item_vii:n`  
`\__enumext_starred_joined_item_viii:n`

The functions `\__enumext_starred_joined_item_vii:n` and `\__enumext_starred_joined_item_viii:n` will set the *width* of the box in which the *content* passed to `\item(<columns>)` will be stored together with the value of `\itemwidth` for the `enumext*` environment.

```

4416 \cs_new_protected:Npn __enumext_starred_joined_item_vii:n #1
4417 {
4418 \int_set:Nn __enumext_joined_item_vii_int {#1}
4419 \int_compare:nNnT { __enumext_joined_item_vii_int } > { __enumext_columns_vii_int }
4420 {
4421 \msg_warning:nnee { enumext } { item-joined }
4422 { \int_use:N __enumext_joined_item_vii_int }
4423 { \int_use:N __enumext_columns_vii_int }
4424 \int_set:Nn __enumext_joined_item_vii_int
4425 {
4426 __enumext_columns_vii_int - __enumext_item_column_pos_vii_int + 1
4427 }
4428 }
4429 \int_compare:nNnT
4430 { __enumext_joined_item_vii_int }
4431 >
4432 { __enumext_columns_vii_int - __enumext_item_column_pos_vii_int + 1 }
4433 {
4434 \msg_warning:nnee { enumext } { item-joined-columns }
4435 { \int_use:N __enumext_joined_item_vii_int }
4436 {
4437 \int_eval:n
4438 { __enumext_columns_vii_int - __enumext_item_column_pos_vii_int + 1 }
4439 }
4440 \int_set:Nn __enumext_joined_item_vii_int
4441 {
4442 __enumext_columns_vii_int - __enumext_item_column_pos_vii_int + 1
4443 }
4444 }
4445 \int_compare:nNnTF { __enumext_joined_item_vii_int } > { 1 }
4446 {
4447 \int_set_eq:NN __enumext_joined_item_aux_vii_int __enumext_joined_item_vii_int

```

```

4448 \int_decr:N \l__enumext_joined_item_aux_vii_int
4449 \int_add:Nn \l__enumext_item_column_pos_vii_int { \l__enumext_joined_item_aux_vii_int }
4450 \int_gadd:Nn \g__enumext_item_count_all_vii_int { \l__enumext_joined_item_aux_vii_int }
4451 \dim_set:Nn \l__enumext_joined_width_vii_dim
4452 {
4453 \l__enumext_item_width_vii_dim * \l__enumext_joined_item_vii_int
4454 + (\l__enumext_labelwidth_vii_dim + \l__enumext_labelsep_vii_dim
4455 + \l__enumext_columns_sep_vii_dim
4456)*\l__enumext_joined_item_aux_vii_int
4457 }
4458 \dim_set_eq:NN \itemwidth \l__enumext_joined_width_vii_dim
4459 }
4460 {
4461 \dim_set_eq:NN \l__enumext_joined_width_vii_dim \l__enumext_item_width_vii_dim
4462 \dim_set_eq:NN \itemwidth \l__enumext_item_width_vii_dim
4463 }
4464 }

```

Same implementation for the `keyans*` environment.

```

4465 \cs_new_protected:Npn __enumext_starred_joined_item_viii:n #1
4466 {
4467 \int_set:Nn \l__enumext_joined_item_viii_int {#1}
4468 \int_compare:nNnT { \l__enumext_joined_item_viii_int } > { \l__enumext_columns_viii_int }
4469 {
4470 \msg_warning:nnee { enumext } { item-joined }
4471 { \int_use:N \l__enumext_joined_item_viii_int }
4472 { \int_use:N \l__enumext_columns_viii_int }
4473 \int_set:Nn \l__enumext_joined_item_viii_int
4474 {
4475 \l__enumext_columns_viii_int - \l__enumext_item_column_pos_viii_int + 1
4476 }
4477 }
4478 \int_compare:nNnT
4479 { \l__enumext_joined_item_viii_int }
4480 >
4481 { \l__enumext_columns_viii_int - \l__enumext_item_column_pos_viii_int + 1 }
4482 {
4483 \msg_warning:nnee { enumext } { item-joined-columns }
4484 { \int_use:N \l__enumext_joined_item_viii_int }
4485 {
4486 \int_eval:n
4487 { \l__enumext_columns_viii_int - \l__enumext_item_column_pos_viii_int + 1 }
4488 }
4489 \int_set:Nn \l__enumext_joined_item_viii_int
4490 {
4491 \l__enumext_columns_viii_int - \l__enumext_item_column_pos_viii_int + 1
4492 }
4493 }
4494 \int_compare:nNnTF { \l__enumext_joined_item_viii_int } > { 1 }
4495 {
4496 \int_set_eq:NN \l__enumext_joined_item_aux_viii_int \l__enumext_joined_item_viii_int
4497 \int_decr:N \l__enumext_joined_item_aux_viii_int
4498 \int_add:Nn \l__enumext_item_column_pos_viii_int { \l__enumext_joined_item_aux_viii_int }
4499 \int_gadd:Nn \g__enumext_item_count_all_viii_int { \l__enumext_joined_item_aux_viii_int }
4500 \dim_set:Nn \l__enumext_joined_width_viii_dim
4501 {
4502 \l__enumext_item_width_viii_dim * \l__enumext_joined_item_viii_int
4503 + (\l__enumext_labelwidth_viii_dim + \l__enumext_labelsep_viii_dim
4504 + \l__enumext_columns_sep_viii_dim
4505)*\l__enumext_joined_item_aux_viii_int
4506 }
4507 \dim_set_eq:NN \itemwidth \l__enumext_joined_width_viii_dim
4508 }
4509 {
4510 \dim_set_eq:NN \l__enumext_joined_width_viii_dim \l__enumext_item_width_viii_dim
4511 \dim_set_eq:NN \itemwidth \l__enumext_item_width_viii_dim
4512 }
4513 }

```

(End of definition for `\__enumext_starred_joined_item_vii:n` and `\__enumext_starred_joined_item_viii:n`)

### 13.43.3 Functions for mini-env, mini-right and mini-right\* keys

`\__enumext_start_mini_vii:` The implementation of the `mini-env` key support is almost identical to the one used in the `enumext` and `keyans` environments, the difference is that the `\__enumext_mini_page` environment on the “right side” is executed “after” closing the environment, so it is necessary to make a global copy of the variable `\l__enumext_minipage_right_vii_dim` in the variable `\g__enumext_minipage_right_vii_dim`.

```

4514 \cs_new_protected:Nn __enumext_start_mini_vii:
4515 {
4516 \dim_compare:nNnT { \l__enumext_minipage_right_vii_dim } > { \c_zero_dim }
4517 {
4518 \dim_set:Nn \l__enumext_minipage_left_vii_dim
4519 {
4520 \linewidth
4521 - \l__enumext_minipage_right_vii_dim
4522 - \l__enumext_minipage_hsep_vii_dim
4523 }
4524 \bool_set_true:N \l__enumext_minipage_active_vii_bool
4525 \dim_gset_eq:NN
4526 \g__enumext_minipage_right_vii_dim
4527 \l__enumext_minipage_right_vii_dim
4528 __enumext_mini_addvspace_vii:
4529 \nointerlineskip\noindent
4530 __enumext_mini_page{ \l__enumext_minipage_left_vii_dim }
4531 }
4532 }

```

The function `\__enumext_stop_mini_vii:` closes the `\__enumext_mini_page` environment on the “left side”, applies `\hfill` and set the variable `\g__enumext_minipage_active_vii_bool` to “true” which will be used in the function `\__enumext_after_env:nn` to execute the `minipage` on the “right side”. At this point we will execute the `\__enumext_stop_list:` and `\__enumext_stop_store_level_vii:` functions stopping the `list` environment and the level saving mechanism for storage in *sequence* of the `\anskey` command and `anskey*` environment. This function is passed to the `\__enumext_after_list_vii:` function in the second part of the `enumext*` environment definition (§13.44).

```

4533 \cs_new_protected:Nn __enumext_stop_mini_vii:
4534 {
4535 \bool_if:NTF \l__enumext_minipage_active_vii_bool
4536 {
4537 __enumext_stop_list:
4538 __enumext_stop_store_level_vii:
4539 \IfDocumentMetadataTF { \tag_resume:n {enumext*} } { }
4540 \end__enumext_mini_page
4541 \hfill
4542 \bool_gset_true:N \g__enumext_minipage_active_vii_bool
4543 }
4544 {
4545 __enumext_stop_list:
4546 __enumext_stop_store_level_vii:
4547 }
4548 }

```

(End of definition for `\__enumext_start_mini_vii:` and `\__enumext_stop_mini_vii:`.)

Finally we execute the `{\code}` passed to the `mini-right` or `mini-right*` keys stored in the variable `\g__enumext_miniright_code_vii_tl` in the `minipage` environment on the “right side”. For compatibility with the `caption` package and possibly other `{\code}` passed to this key, we will pass it to a box and then print it.

```

4549 __enumext_after_env:nn {enumext*}
4550 {
4551 \bool_if:NT \g__enumext_minipage_active_vii_bool
4552 {
4553 __enumext_minipage:w [t] { \g__enumext_minipage_right_vii_dim }
4554 \legacy_if_gset_false:n { @minipage }
4555 \skip_vertical:N \c_zero_skip
4556 \par\addvspace { \g__enumext_minipage_right_skip }
4557 \bool_if:NF \g__enumext_minipage_center_vii_bool
4558 {
4559 \tl_put_left:Nn \g__enumext_miniright_code_vii_tl
4560 {
4561 \centering
4562 }
4563 }
4564 \vbox_set_top:Nn \l__enumext_miniright_code_vii_box

```

```

4565 {
4566 \tl_use:N \g__enumext_miniright_code_vii_tl
4567 }
4568 \box_use_drop:N \l__enumext_miniright_code_vii_box
4569 \skip_vertical:N \c_zero_skip
4570 __enumext_endminipage:
4571 \par\addvspace{ \g__enumext_minipage_after_skip }
4572 }
4573 \bool_gset_false:N \g__enumext_minipage_active_vii_bool
4574 \bool_gset_true:N \g__enumext_minipage_center_vii_bool
4575 \tl_gclear:N \g__enumext_miniright_code_vii_tl
4576 \dim_gzero:N \g__enumext_minipage_right_vii_dim
4577 \bool_gset_false:N \g__enumext_starred_bool
4578 }

```

```

__enumext_start_mini_viii:
__enumext_stop_mini_viii:

```

The implementation of the `mini-env`, `mini-right` and `mini-right*` keys is identical to the one used in the `enumext*` environment.

```

4579 \cs_new_protected:Nn __enumext_start_mini_viii:
4580 {
4581 \dim_compare:nNtT { \l__enumext_minipage_right_viii_dim } > { \c_zero_dim }
4582 {
4583 \dim_set:Nn \l__enumext_minipage_left_viii_dim
4584 {
4585 \linewidth
4586 - \l__enumext_minipage_right_viii_dim
4587 - \l__enumext_minipage_hsep_viii_dim
4588 }
4589 \bool_set_true:N \l__enumext_minipage_active_viii_bool
4590 \dim_gset_eq:NN
4591 \g__enumext_minipage_right_viii_dim
4592 \l__enumext_minipage_right_viii_dim
4593 __enumext_mini_addvspace_viii:
4594 \nointerlineskip\noindent
4595 __enumext_mini_page{ \l__enumext_minipage_left_viii_dim }
4596 }
4597 }
4598 \cs_new_protected:Nn __enumext_stop_mini_viii:
4599 {
4600 \bool_if:NTF \l__enumext_minipage_active_viii_bool
4601 {
4602 __enumext_stop_list:
4603 \IfDocumentMetadataTF { \tag_resume:n {keyans*} } { } { }
4604 \end__enumext_mini_page
4605 \hfill
4606 \bool_gset_true:N \g__enumext_minipage_active_viii_bool
4607 }
4608 {
4609 __enumext_stop_list:
4610 }
4611 }
4612 __enumext_after_env:nn {keyans*}
4613 {
4614 \bool_if:NT \g__enumext_minipage_active_viii_bool
4615 {
4616 __enumext_mini_page{ \g__enumext_minipage_right_viii_dim }
4617 \par\addvspace { \g__enumext_minipage_right_skip }
4618 \bool_if:NF \g__enumext_minipage_center_viii_bool
4619 {
4620 \tl_put_left:Nn \g__enumext_miniright_code_viii_tl
4621 {
4622 \centering
4623 }
4624 }
4625 \vbox_set_top:Nn \l__enumext_miniright_code_viii_box
4626 {
4627 \tl_use:N \g__enumext_miniright_code_viii_tl
4628 }
4629 \box_use_drop:N \l__enumext_miniright_code_viii_box
4630 \end__enumext_mini_page
4631 \par\addvspace{ \g__enumext_minipage_after_skip }
4632 }

```

```

4633 \bool_gset_false:N \g__enumext_minipage_active_viii_bool
4634 \bool_gset_true:N \g__enumext_minipage_center_viii_bool
4635 \tl_gclear:N \g__enumext_miniright_code_viii_tl
4636 \dim_gzero:N \g__enumext_minipage_right_viii_dim
4637 }

```

(End of definition for \\_\_enumext\_start\_mini\_viii: and \\_\_enumext\_stop\_mini\_viii:.)

### 13.44 The environment enumext\*

**enumext\*** First we will generate the environment and we will give a temporary definition to \\_\_enumext\_stop\_item\_tmp\_vii: equal to \\_\_enumext\_first\_item\_tmp\_vii: and next to \item equal to \\_\_enumext\_start\_item\_tmp\_vii: which we will redefine later. Unlike the implementation used by the **shortlst** package, we will not set the values of \rightskip and \@rightskip equal to \@flushglue whose value is 0.0pt plus 1.0 fil, in the tests I have performed this fails in some circumstances and different results are obtained when using pdfTeX and LuaTeX.

```

4638 \NewDocumentEnvironment{enumext*}{o}{
4639 {
4640 __enumext_safe_exec_vii:
4641 __enumext_parse_keys_vii:n {#1}
4642 __enumext_before_list_vii:
4643 __enumext_start_store_level_vii:
4644 __enumext_start_list:nn { }
4645 {
4646 __enumext_list_arg_two_vii:
4647 __enumext_before_keys_exec_vii:
4648 }
4649 \IfDocumentMetadataTF { \tag_suspend:n {enumext*} } { }
4650 __enumext_starred_columns_set_vii:
4651 \item[] \scan_stop:
4652 \cs_set_eq:NN __enumext_stop_item_tmp_vii: __enumext_first_item_tmp_vii:
4653 \cs_set_eq:NN \item __enumext_start_item_tmp_vii:
4654 \ignorespaces
4655 }
4656 {
4657 \IfDocumentMetadataTF { \tag_struct_end:n {tag=text-unit} } { }
4658 __enumext_stop_item_tmp_vii:
4659 __enumext_remove_extra_parsep_vii:
4660 __enumext_after_list_vii:
4661 }
}

```

(End of definition for enumext\*. This function is documented on page 5.)

\\_\_enumext\_safe\_exec\_vii: We will first call the function \\_\_enumext\_is\_not\_nested: which sets \g\_\_enumext\_starred\_bool to true if we are NOT nested within **enumext**, then call the function \\_\_enumext\_internal\_mini\_page: to create the environment **\_\_enumext\_mini\_page**, we will increment \l\_\_enumext\_level\_h\_int to restrict nesting of the environment, set \l\_\_enumext\_starred\_bool to true and finally call the function \\_\_enumext\_is\_on\_first\_level: which sets \l\_\_enumext\_starred\_first\_bool to true if we are not nested, allowing the “storage system” to be used.

```

4662 \cs_new_protected:Nn __enumext_safe_exec_vii:
4663 {
4664 __enumext_is_not_nested:
4665 __enumext_internal_mini_page:
4666 \int_incr:N \l__enumext_level_h_int
4667 \int_compare:nNt { \l__enumext_level_h_int } > { 1 }
4668 {
4669 \msg_error:nn { enumext } { nested }
4670 }
4671 \int_compare:nNt { \l__enumext_keyans_level_h_int } = { 1 }
4672 {
4673 \msg_error:nnn { enumext } { nested-horizontal } { keyans*}
4674 }
4675 \bool_set_true:N \l__enumext_starred_bool
4676 \bool_set_false:N \l__enumext_standar_bool
4677 __enumext_is_on_first_level:
4678 }

```

(End of definition for \\_\_enumext\_safe\_exec\_vii:.)

\\_\_enumext\_parse\_keys\_vii:n

First we will clear the variable \l\_\_enumext\_series\_str used by the key `series`, process the environment [`<key = val>`] and execute the function \\_\_enumext\_parse\_series:n and used by the key `series`, then we execute the function \\_\_enumext\_store\_active\_keys\_vii:n and reprocess the `<keys>` to pass them to the storage `sequence` if the key `save-key` is not active.

```

4679 \cs_new_protected:Npn __enumext_parse_keys_vii:n #1
4680 {
4681 \tl_if_novalue:nF {#1}
4682 {
4683 \str_clear:N \l__enumext_series_str
4684 \keys_set:nn { enumext / enumext* } {#1}
4685 __enumext_parse_series:n {#1}
4686 __enumext_store_active_keys_vii:n {#1}
4687 }
4688 }
```

(End of definition for \\_\_enumext\_parse\_keys\_vii:n.)

\\_\_enumext\_before\_list\_vii:

The function \\_\_enumext\_before\_list\_vii: first calls the function \\_\_enumext\_vspace\_above\_vii: used by the keys `above` and `above*`, then calls the function \\_\_enumext\_check\_ans\_active: for the check answer mechanism and finally calls the functions \\_\_enumext\_before\_args\_exec: and \\_\_enumext\_start\_mini\_vii: used by the keys `before*`, `mini-env`, `mini-right` and `mini-right*`.

```

4689 \cs_new_protected:Nn __enumext_before_list_vii:
4690 {
4691 __enumext_vspace_above_vii:
4692 __enumext_check_ans_active:
4693 __enumext_before_args_exec_vii:
4694 __enumext_start_mini_vii:
4695 }
```

(End of definition for \\_\_enumext\_before\_list\_vii:.)

\\_\_enumext\_after\_list\_vii:

The function \\_\_enumext\_after\_list\_vii: first calls the function \\_\_enumext\_stop\_mini\_vii: which internally calls \\_\_enumext\_stop\_list: and \\_\_enumext\_stop\_store\_level\_vii: (§13.43.3) used by the keys `mini-env`, `mini-right` and `mini-right*`, then to the functions \\_\_enumext\_after\_stop\_list\_vii: used by the key `after`, \\_\_enumext\_check\_ans\_key\_hook: used by the key `check-ans`, \\_\_enumext\_vspace\_below\_vii: used by the keys `below` and `below*`. Finally set \l\_\_enumext\_starred\_bool to false and call the \\_\_enumext\_resume\_save\_counter: function used by the `series`, `resume` and `resume*` keys.

```

4696 \cs_new_protected:Nn __enumext_after_list_vii:
4697 {
4698 __enumext_stop_mini_vii:
4699 __enumext_after_stop_list_vii:
4700 __enumext_check_ans_key_hook:
4701 __enumext_vspace_below_vii:
4702 \bool_set_false:N \l__enumext_starred_bool
4703 __enumext_resume_save_counter:
4704 }
```

(End of definition for \\_\_enumext\_after\_list\_vii:.)

\\_\_enumext\_start\_store\_level\_vii:

\\_\_enumext\_stop\_store\_level\_vii:

The \\_\_enumext\_start\_store\_level\_vii: and \\_\_enumext\_stop\_store\_level\_vii: functions activate the “*storing structure*” mechanism in *sequence* for \anskey command and `anskey*` environment if `enumext*` are nested in `enumext`.

```

4705 \cs_new_protected:Nn __enumext_start_store_level_vii:
4706 {
4707 \bool_if:NT \l__enumext_store_active_bool
4708 {
4709 \int_compare:nNt { \l__enumext_level_int } > { 0 }
4710 {
4711 __enumext_store_level_open_vii:
4712 }
4713 }
4714 }
4715 \cs_new_protected:Nn __enumext_stop_store_level_vii:
4716 {
4717 \bool_if:NT \l__enumext_store_active_bool
4718 {
4719 \int_compare:nNt { \l__enumext_level_int } > { 0 }
4720 {
4721 __enumext_store_level_close_vii:

```



```

4722 }
4723 }
4724 }

```

(End of definition for `\__enumext_start_store_level_vii:` and `\__enumext_stop_store_level_vii:`.)

### 13.44.1 The command `\item` in `enumext*`

`\__enumext_first_item_tmp_vii:`

The `\__enumext_first_item_tmp_vii:` function will remove horizontal space equal to `\labelwidth` plus `\labelsep` to the left of the “*first*” `\item` in the environment at the point of execution of this function, where it is equal to the `\__enumext_stop_item_tmp_vii:` function inside the environment body definition.

```

4725 \cs_new_protected_nopar:Nn __enumext_first_item_tmp_vii:
4726 {
4727 \skip_horizontal:n
4728 {
4729 -__enumext_labelwidth_vii_dim - __enumext_labelsep_vii_dim
4730 }
4731 \ignorespaces
4732 }

```

(End of definition for `\__enumext_first_item_tmp_vii:`.)

`\__enumext_start_item_tmp_vii:`

`\__enumext_item_peek_args_vii:`

`\__enumext_joined_item_vii:w`

`\__enumext_standar_item_vii:w`

`\__enumext_starred_item_vii:w`

First we will call the function `\__enumext_stop_item_tmp_vii:` that we will redefine later, we will increment the value of `\__enumext_item_column_pos_vii_int` that will count the item’s by rows and the value of `\g__enumext_item_count_all_vii_int` that will count the total of item’s in the environment. After that we will call the function `\__enumext_item_peek_args_vii:` that will handle the arguments passed to `\item`.

```

4733 \cs_new_protected_nopar:Nn __enumext_start_item_tmp_vii:
4734 {
4735 __enumext_stop_item_tmp_vii:
4736 \int_incr:N __enumext_item_column_pos_vii_int
4737 \int_gincr:N \g__enumext_item_count_all_vii_int
4738 __enumext_item_peek_args_vii:
4739 }

```

The function `\__enumext_item_peek_args_vii:` will handle the `\item(<number>)`. Look for the argument “(”, if it is present we will call the function `\__enumext_joined_item_vii:w (<number>)`, which is in charge of joining the item’s in the same row, in case they are not present we will set the default value (1).

```

4740 \cs_new_protected:Nn __enumext_item_peek_args_vii:
4741 {
4742 \peek_meaning:NTF (
4743 { __enumext_joined_item_vii:w }
4744 { __enumext_joined_item_vii:w (1) }
4745 }

```

The function `\__enumext_joined_item_vii:w` will first call the function `\__enumext_starred_joined_item_vii:n` in charge of setting the *width* of the box that will store the content passed to `\item`. Then we will look for the argument “\*”, if it is present we will call the function `\__enumext_starred_item_vii:w` otherwise we will call the function `\__enumext_standar_item_vii:w`.

```

4746 \cs_new_protected:Npn __enumext_joined_item_vii:w (#1)
4747 {
4748 __enumext_starred_joined_item_vii:n {#1}
4749 \peek_meaning_remove:NTF *
4750 { __enumext_starred_item_vii:w }
4751 { __enumext_standar_item_vii:w }
4752 }

```

The function `\__enumext_standar_item_vii:w` will first look for the argument “[”, if present it will set the state of the variable `\__enumext_wrap_label_opt_vii_bool` equal to the state of the variable `\l__enumext_wrap_label_opt_vii_bool` handled by the key `wrap-label*` and finally execute the *non-enumerated* version `\item[<custom>]` by means of the function `\__enumext_start_item_vii:w`, otherwise we will set the value of the variable `\l__enumext_wrap_label_vii_bool` handled by the `wrap-label` key to true and set the switch `\if@noitemarg` to true to execute the enumerated version of `\item` by means of the function `\__enumext_start_item_vii:w [ \__enumext_label_vii_tl ]`.

```

4753 \cs_new_protected:Npn __enumext_standar_item_vii:w
4754 {
4755 \bool_set_false:N \l__enumext_item_starred_vii_bool
4756 \peek_meaning:NTF [
4757 {
4758 \bool_set_eq:NN \l__enumext_wrap_label_vii_bool \l__enumext_wrap_label_opt_vii_bool
4759 __enumext_start_item_vii:w
4760 }

```

```

4761 {
4762 \bool_set_true:N \l__enumext_wrap_label_vii_bool
4763 \legacy_if_set_true:n { @noitemarg }
4764 __enumext_start_item_vii:w [\l__enumext_label_vii_tl] \ignorespaces
4765 }
4766 }

```

The function `\__enumext_starred_item_vii:w` together with the specified auxiliary functions `aux_i:w`, `aux_ii:w`, and `aux_iii:w` execute `\item*`, `\item*[\langle symbol \rangle]` and `\item*[\langle symbol \rangle][\langle offset \rangle]`.

```

4767 \cs_new_protected:Npn __enumext_starred_item_vii:w
4768 {
4769 \bool_set_true:N \l__enumext_item_starred_vii_bool
4770 \bool_set_true:N \l__enumext_wrap_label_vii_bool
4771 \peek_meaning:NTF [
4772 { __enumext_starred_item_vii_aux_i:w }
4773 { __enumext_starred_item_vii_aux_ii:w }
4774]
4775 \cs_new_protected:Npn __enumext_starred_item_vii_aux_i:w [#1]
4776 {
4777 \tl_gset:Nn \g__enumext_item_symbol_aux_vii_tl {#1}
4778 __enumext_starred_item_vii_aux_ii:w
4779 }
4780 \cs_new_protected:Npn __enumext_starred_item_vii_aux_ii:w
4781 {
4782 \peek_meaning:NTF [
4783 { __enumext_starred_item_vii_aux_iii:w }
4784 {
4785 \dim_set_eq:NN \l__enumext_item_symbol_sep_vii_dim \l__enumext_labelsep_vii_dim
4786 \legacy_if_set_true:n { @noitemarg }
4787 __enumext_start_item_vii:w [\l__enumext_label_vii_tl] \ignorespaces
4788 }
4789]
4790 \cs_new_protected:Npn __enumext_starred_item_vii_aux_iii:w [#1]
4791 {
4792 \dim_set:Nn \l__enumext_item_symbol_sep_vii_dim {#1}
4793 \legacy_if_set_true:n { @noitemarg }
4794 __enumext_start_item_vii:w [\l__enumext_label_vii_tl] \ignorespaces
4795 }

```

(End of definition for `\__enumext_start_item_tmp_vii:` and others.)

`\__enumext_fake_make_label_vii:n`

The `\__enumext_fake_make_label_vii:n` function will be in charge of handling our definition of `\item`. First we increment the counter `enumXvii` for the enumerated items and activate support for the *check answers* mechanism, followed by support for `\item*[\langle symbol \rangle][\langle offset \rangle]` if present, then the `wrap-label` and `wrap-label*` keys which we execute using `\makebox` whose width will be given by the `labelwidth` key and position by the `align` key, inside the argument of this we will execute the `font` key together with the function defined by the `wrap-label` or `wrap-label*` keys. Finally we execute the `labelsep` key applying a `\skip_horizontal:N` and `\ignorespaces`.

💡 For compatibility with *tagged* PDF and *hyperref* when an environment `enumext` is nested in `enumext*` and the key `save-ans` is not active need setting the `\if@hyper@item` switch to “true”. The explanation for this is given by the master Heiko Oberdiek on `\refstepcounter{enumi}` twice (or more) creates destination with the same identifier. This patch is only needed if you are running `pdflatex` and not if you are running `lua1latex`

```

4796 \cs_new_protected_nopar:Npn __enumext_fake_make_label_vii:n #1
4797 {
4798 \legacy_if:nT { @noitemarg }
4799 {
4800 \legacy_if_set_false:n { @noitemarg }
4801 \legacy_if:nT { @nmbrrlist }
4802 {
4803 \IfDocumentMetadataTF
4804 {
4805 \bool_if:NT \l__enumext_hyperref_bool
4806 {
4807 \legacy_if_set_true:n { @hyper@item }
4808 }
4809 } { }
4810 \refstepcounter{enumXvii}
4811 \bool_if:NT \l__enumext_check_answers_bool
4812 {
4813 \int_gincr:N \g__enumext_item_number_int

```

```

4814 \bool_set_true:N \l__enumext_item_number_bool
4815 }
4816 }
4817 }
4818 \bool_if:NT \l__enumext_item_starred_vii_bool
4819 {
4820 \tl_if_blank:VT \g__enumext_item_symbol_aux_vii_tl
4821 {
4822 \tl_gset_eq:NN
4823 \g__enumext_item_symbol_aux_vii_tl \l__enumext_item_symbol_vii_tl
4824 }
4825 \mode_leave_vertical:
4826 \skip_horizontal:n { -\l__enumext_item_symbol_sep_vii_dim }
4827 \hbox_overlap_left:n { \g__enumext_item_symbol_aux_vii_tl }
4828 \skip_horizontal:N \l__enumext_item_symbol_sep_vii_dim
4829 \tl_gclear:N \g__enumext_item_symbol_aux_vii_tl
4830 }
4831 \makebox[\l__enumext_labelwidth_vii_dim][\l__enumext_align_label_vii_str]
4832 {
4833 \tl_use:N \l__enumext_label_font_style_vii_tl
4834 \bool_if:NTF \l__enumext_wrap_label_vii_bool
4835 {
4836 __enumext_wrapper_label_vii:n {#1}
4837 }
4838 { #1 }
4839 }
4840 \skip_horizontal:N \l__enumext_labelsep_vii_dim \ignorespaces
4841 }

```

(End of definition for \\_\_enumext\_fake\_make\_label\_vii:n.)

### 13.44.2 Real definition of \item in enumext\*

The functions \\_\_enumext\_start\_item\_vii:w and \\_\_enumext\_stop\_item\_vii: executing the true definition of \item inside the enumext\* environment, unlike the implementation in shortlst we will NOT use an extra group and the plain form of the lrbox environment.

```

__enumext_start_item_vii:w
__enumext_stop_item_vii:

```

The first thing we will do is set the value of \\_\_enumext\_stop\_item\_tmp\_vii: equal to \\_\_enumext\_stop\_item\_vii: which we will define later, after that we will start capturing \item and “item content” in a horizontal box where the width will be \itemwidth plus \labelwidth plus \labelsep.

```

4842 \cs_new_protected_nopar:Npn __enumext_start_item_vii:w [#1]
4843 {
4844 \cs_set_eq:NN __enumext_stop_item_tmp_vii: __enumext_stop_item_vii:
4845 \hbox_set_to_wd:Nnw \l__enumext_item_text_vii_box
4846 {
4847 \l__enumext_joined_width_vii_dim
4848 + \l__enumext_labelwidth_vii_dim
4849 + \l__enumext_labelsep_vii_dim
4850 }

```

Redefine the \footnote command.

```

4851 __enumext_renew_footnote_starred:

```

Now we insert our sockets for tagging PDF support and run \item.

```

4852 __enumext_start_list_tag:n {enumext*}
4853 __enumext_fake_make_label_vii:n {#1}
4854 __enumext_stop_start_list_tag:

```

Finally we open the minipage environment, capture the “item content”, make \parindent take the value of the key listparindent and \parskip take the value of the key parsep, then execute the keys itemindent and first.

- Here the use of \unskip and \skip\_horizontal:n with the value of listparindent is necessary, otherwise an unwanted space is created when using \item[⟨opt⟩] and the value passed to the key itemindent is incremented.

```

4855 __enumext_minipage:w [t]{ \l__enumext_joined_width_vii_dim }
4856 \dim_set_eq:NN \parindent \l__enumext_listparindent_vii_dim
4857 \skip_set_eq:NN \parskip \l__enumext_parsep_vii_skip
4858 __enumext_unskip_unkern:
4859 __enumext_unskip_unkern:
4860 \skip_horizontal:n { -\l__enumext_listparindent_vii_dim } \ignorespaces
4861 \tl_use:N \l__enumext_fake_item_indent_vii_tl
4862 \tl_use:N \l__enumext_after_list_args_vii_tl
4863 }

```

The `\__enumext_stop_item_vii:` function will finish the fetching `\item` and “*item content*” by closing the `minipage` environment, the *sockets for tagging PDF* and the *horizontal box*.

```
4864 \cs_new_protected_nopar:Nn __enumext_stop_item_vii:
4865 {
4866 __enumext_endminipage:
4867 __enumext_stop_list_tag:n {enumext*}
4868 \hbox_set_end:
```

Here we will reduce the *warnings* a bit by setting the value of `\hbadness` to `10000`, print `\item` and “*item content*” from the *horizontal box*.

```
4869 \int_set:Nn \hbadness { 10000 }
4870 \box_use_drop:N \l__enumext_item_text_vii_box
```

Finally apply the *vertical space* between rows set by `itemsep` key passed to `\parsep` using `\par\noindent` and *horizontal space* between columns set by `columns-sep` key using `\skip_horizontal:N`.

```
4871 \int_compare:nNnTF
4872 { \l__enumext_item_column_pos_vii_int } = { \l__enumext_columns_vii_int }
4873 {
4874 \par\noindent
4875 \int_zero:N \l__enumext_item_column_pos_vii_int
4876 }
4877 {
4878 \skip_horizontal:N \l__enumext_columns_sep_vii_dim
4879 }
4880 }
```

(End of definition for `\__enumext_start_item_vii:w` and `\__enumext_stop_item_vii:.`)

`\__enumext_remove_extra_parsep_vii:`

Remove the extra *vertical space* equal to `\parsep=\itemsep` when the total number of `\item` is divisible by the number of `\item` in the last row of the environment. Here the use of `\unskip` or `\removeatlastskip` fails and does not obtain the expected result, using `\vspace` is the option and in this case, we can use a simplified version since we are always in *(vertical mode)*.

```
4881 \cs_new_protected:Nn __enumext_remove_extra_parsep_vii:
4882 {
4883 \int_compare:nNnT
4884 {
4885 \int_mod:nn
4886 { \g__enumext_item_count_all_vii_int } { \l__enumext_columns_vii_int }
4887 }
4888 =
4889 { 0 }
4890 {
4891 \para_end:
4892 \skip_vertical:n { -\l__enumext_itemsep_vii_skip }
4893 \skip_vertical:N \c_zero_skip
4894 \int_gzero:N \g__enumext_item_count_all_vii_int
4895 }
4896 }
```

(End of definition for `\__enumext_remove_extra_parsep_vii:.`)

As we don’t want our check to be executed `check-ans` by levels but on the complete list, we will take it out of the `enumext*` environment using the “*hook*” function `\__enumext_after_env:nn`.

```
4897 __enumext_after_env:nn {enumext*}
4898 {
4899 __enumext_execute_after_env:
4900 }
```

### 13.45 The environment `keyans*`

`keyans*`

The implementation of `keyans*` environment is the similar as that used by the `enumext*` environment except for the `\__enumext_check_starred_cmd:n` function added in the second part.

```
4901 \NewDocumentEnvironment{keyans*}{ o }
4902 {
4903 __enumext_safe_exec_viii:
4904 __enumext_parse_keys_viii:n {#1}
4905 __enumext_before_list_viii:
4906 __enumext_start_list:nn { }
4907 {
4908 __enumext_list_arg_two_viii:
4909 __enumext_before_keys_exec_viii:
4910 }
```

```

4911 \IfDocumentMetadataTF { \tag_suspend:n {keyans*} } { }
4912 __enumext_starred_columns_set_viii:
4913 \item[] \scan_stop:
4914 \cs_set_eq:NN __enumext_stop_item_tmp_viii: __enumext_first_item_tmp_viii:
4915 \cs_set_eq:NN \item __enumext_start_item_tmp_viii:
4916 \ignorespaces
4917 }
4918 {
4919 \IfDocumentMetadataTF { \tag_struct_end:n {tag=text-unit} } { }
4920 __enumext_stop_item_tmp_viii:
4921 __enumext_remove_extra_parsep_viii:
4922 __enumext_check_starred_cmd:n { item }
4923 __enumext_after_list_viii:
4924 }

```

(End of definition for `keyans*`. This function is documented on page 14.)

`\__enumext_safe_exec_viii:` The `\__enumext_safe_exec_viii:` function will first check if the `save-ans` key is active and only when this is true the environment will be available, it will increment the value of `\__enumext_keyans_level_h_int` and return an error message when we are nesting the environment, then it will call the `\__enumext_keyans_name_and_start:` function in charge of saving the name of the environment and the line it is running on, then it will check if we are trying to nest `keyans*` in `enumext*` returning an error and we will set `\__enumext_starred_bool` to true, finally we will check if we are within the appropriate level within the `enumext` environment.

```

4925 \cs_new_protected:Nn __enumext_safe_exec_viii:
4926 {
4927 \bool_if:NF __enumext_store_active_bool
4928 {
4929 \msg_error:nnnn { enumext } { wrong-place } { keyans* } { save-ans }
4930 }
4931 \int_incr:N __enumext_keyans_level_h_int
4932 \int_compare:nNnT { __enumext_keyans_level_h_int } > { 1 }
4933 {
4934 \msg_error:nn { enumext } { nested }
4935 }
4936 __enumext_keyans_name_and_start:
4937 \bool_if:NT __enumext_starred_bool
4938 {
4939 \msg_error:nnn { enumext } { nested-horizontal } { enumext* }
4940 }
4941 \bool_set_true:N __enumext_starred_bool
4942 % Set false for interfering with enumext nested in keyans* (yes, its possible and crayze)
4943 \bool_set_false:N __enumext_store_active_bool
4944 \int_compare:nNnT { __enumext_level_int } > { 1 }
4945 {
4946 \msg_error:nn { enumext } { keyans-wrong-level }
4947 }
4948 }

```

(End of definition for `\__enumext_safe_exec_viii:`.)

`\__enumext_parse_keys_viii:n` Parse [`<key = val>`] for `keyans*`.

```

4949 \cs_new_protected:Npn __enumext_parse_keys_viii:n #1
4950 {
4951 \tl_if_novalue:nF {#1}
4952 {
4953 \keys_set:nn { enumext / keyans* } {#1}
4954 }
4955 }

```

(End of definition for `\__enumext_parse_keys_viii:n`.)

`\__enumext_before_list_viii:` The function `\__enumext_before_list_viii:` will add the vertical spacing on the environment if the `above` key is active next to the `{<code>}` defined by the `before*` key if it is active, the call the function `\__enumext_start_mini_viii:` handle by `mini-env`.

```

4956 \cs_new_protected:Nn __enumext_before_list_viii:
4957 {
4958 __enumext_vspace_above_viii:
4959 __enumext_before_args_exec_viii:
4960 __enumext_start_mini_viii:
4961 }

```

(End of definition for `\__enumext_before_list_viii:`)

`\__enumext_after_list_viii:` The function `\__enumext_after_list_viii:` first call the function `\__enumext_stop_mini_viii:`, then apply the `{\code}` handled by the `after` key together with the *vertical space* handled by the `below` key if they are present.

```
4962 \cs_new_protected:Nn __enumext_after_list_viii:
4963 {
4964 __enumext_stop_mini_viii:
4965 __enumext_after_stop_list_viii:
4966 __enumext_vspace_below_viii:
4967 }
```

(End of definition for `\__enumext_after_list_viii:`)

### 13.45.1 The command `\item` in `keyans*`

The idea here is to make the `\item` command behave in the same way as in the `keyans` environment with the difference of the *optional argument* (`\number`) which works in the same way as in the `enumext*` environment. In simple terms we want to store the `\label` next to the `[\content]` if it is present in the *sequence* and *prop list* defined by `save-ans` key for `\item*`, `\item*[\content]`, `\item(\number)*` and `\item(\number)*[\content]` commands.

`\__enumext_first_item_tmp_viii:` The `\__enumext_first_item_tmp_viii:` function will remove horizontal space equal to `\labelwidth` plus `\labelsep` to the left of the “first” `\item` in the environment at the point of execution of this function, where it is equal to the `\__enumext_stop_item_tmp_viii:` function inside the environment body definition.

```
4968 \cs_new_protected_nopar:Nn __enumext_first_item_tmp_viii:
4969 {
4970 \skip_horizontal:n
4971 {
4972 -__enumext_labelwidth_viii_dim - __enumext_labelsep_viii_dim
4973 }
4974 \ignorespaces
4975 }
```

(End of definition for `\__enumext_first_item_tmp_viii:`)

`\__enumext_start_item_tmp_viii:` First we will call the function `\__enumext_stop_item_tmp_viii:` that we will redefine later, we will  
`\__enumext_item_peek_args_viii:` increment the value of `\l__enumext_item_column_pos_viii_int` that will count the item’s by rows and  
`\__enumext_joined_item_viii:w` the value of `\g__enumext_item_count_all_viii_int` that will count the total of item’s in the environment.  
`\__enumext_standar_item_viii:w` After that we will call the function `\__enumext_item_peek_args_viii:` that will handle the arguments passed to `\item`.

```
4976 \cs_new_protected_nopar:Nn __enumext_start_item_tmp_viii:
4977 {
4978 __enumext_stop_item_tmp_viii:
4979 \int_incr:N \l__enumext_item_column_pos_viii_int
4980 \int_gincr:N \g__enumext_item_count_all_viii_int
4981 __enumext_item_peek_args_viii:
4982 }
```

The function `\__enumext_item_peek_args_viii:` will handle the `\item(\number)`. Look for the argument “(”, if it is present we will call the function `\__enumext_joined_item_viii:w(\number)`, which is in charge of joining the item’s in the same row, in case they are not present we will set the default value (1).

```
4983 \cs_new_protected:Nn __enumext_item_peek_args_viii:
4984 {
4985 \peek_meaning:NTF (
4986 { __enumext_joined_item_viii:w }
4987 { __enumext_joined_item_viii:w (1) }
4988 }
```

The function `\__enumext_joined_item_viii:w` will first call the function `\__enumext_starred_joined_item_viii:n` in charge of setting the *width* of the box that will store the content passed to `\item`. Then we will look for the argument “\*”, if it is present we will call the function `\__enumext_starred_item_viii:w` otherwise we will call the function `\__enumext_standar_item_viii:w`.

```
4989 \cs_new_protected:Npn __enumext_joined_item_viii:w (#1)
4990 {
4991 __enumext_starred_joined_item_viii:n {#1}
4992 \peek_meaning_remove:NTF *
4993 { __enumext_starred_item_viii:w }
4994 { __enumext_standar_item_viii:w }
4995 }
```

The function `\__enumext_standar_item_viii:w` will first look for the argument “[”, if present it will set the state of the variable `\l__enumext_wrap_label_opt_viii_bool` equal to the state of the variable `\l__enumext_wrap_label_opt_viii_bool` handled by the key `wrap-label*` and finally execute the *non-enumerated* version `\item[⟨custom⟩]` by means of the function `\__enumext_start_item_viii:w`, otherwise we will set the value of the variable `\l__enumext_wrap_label_viii_bool` handled by the `wrap-label` key to true and set the switch `\if@noitemarg` to true to execute the enumerated version of `\item` by means of the function `\__enumext_start_item_viii:w [ \l__enumext_label_viii_tl ]`.

```

4996 \cs_new_protected:Npn __enumext_standar_item_viii:w
4997 {
4998 \bool_set_false:N \l__enumext_item_starred_viii_bool
4999 \int_zero:N \l__enumext_item_star_exec_int
5000 \peek_meaning:NTF [
5001 {
5002 \bool_set_eq:NN \l__enumext_wrap_label_viii_bool \l__enumext_wrap_label_opt_viii_bool
5003 __enumext_start_item_viii:w
5004 }
5005 {
5006 \bool_set_true:N \l__enumext_wrap_label_viii_bool
5007 \legacy_if_set_true:n { @noitemarg }
5008 __enumext_start_item_viii:w [\l__enumext_label_viii_tl] \ignorespaces
5009 }
5010 }

```

(End of definition for `\__enumext_start_item_tmp_viii:` and others.)

```

__enumext_starred_item_viii:w
__enumext_starred_item_viii_aux_i:w
__enumext_starred_item_viii_aux_ii:w
__enumext_starred_item_exec:

```

The function `\__enumext_starred_item_viii:w` together with the specified auxiliary functions `aux_i:w` and `aux_ii:w` execute `\item*` and `\item*[⟨content⟩]`.

```

5011 \cs_new_protected:Npn __enumext_starred_item_viii:w
5012 {
5013 \bool_set_true:N \l__enumext_item_starred_viii_bool
5014 \int_incr:N \l__enumext_item_star_exec_int
5015 \bool_set_true:N \l__enumext_wrap_label_viii_bool
5016 \peek_meaning:NTF [
5017 { __enumext_starred_item_viii_aux_i:w }
5018 { __enumext_starred_item_viii_aux_ii:w }
5019 }

```

The function `\__enumext_starred_item_viii_aux_i:w` will save the *optional argument* to `\item*` in `\l__enumext_store_current_opt_arg_tl` and will save this argument along with the spacing set by the key `save-sep` in variable `\l__enumext_store_current_label_tl` if present, then call the function `\__enumext_starred_item_viii_aux_ii:w`.

```

5020 \cs_new_protected:Npn __enumext_starred_item_viii_aux_i:w [#1]
5021 {
5022 \tl_clear:N \l__enumext_store_current_label_tl
5023 \tl_if_no_value:nF { #1 }
5024 {
5025 \tl_if_empty:NF \l__enumext_store_keyans_item_opt_sep_tl
5026 {
5027 \tl_put_right:Ne \l__enumext_store_current_label_tl
5028 {
5029 \l__enumext_store_keyans_item_opt_sep_tl
5030 }
5031 \tl_put_right:Ne \l__enumext_store_current_label_tl { #1 }
5032 }
5033 \tl_set:Ne \l__enumext_store_current_opt_arg_tl { #1 }
5034 }
5035 __enumext_starred_item_viii_aux_ii:w
5036 }
5037 \cs_new_protected:Npn __enumext_starred_item_viii_aux_ii:w
5038 {
5039 \legacy_if_set_true:n { @noitemarg }
5040 __enumext_start_item_viii:w [\l__enumext_label_viii_tl] \ignorespaces
5041 }

```

The function `\__enumext_starred_item_exec:` will be in charge of storing the current *⟨label⟩* for `\item*` followed by the *⟨⟨content⟩⟩* for `\item*[⟨content⟩]` if present in the *sequence* and *prop list* set by the `save-ans` key. In this same function the keys `show-ans`, `show-pos` and `save-ref` are implemented.

```

5042 \cs_new_protected:Nn __enumext_starred_item_exec:
5043 {
5044 \tl_put_left:Ne \l__enumext_store_current_label_tl { \l__enumext_label_viii_tl }
5045 __enumext_store_addto_prop:V \l__enumext_store_current_label_tl

```



```

5046 __enumext_keyans_store_ref:
5047 \tl_put_left:Ne \l__enumext_store_current_label_tl { \item }
5048 __enumext_keyans_addto_seq_link:
5049 \int_gincr:N \g__enumext_check_starred_cmd_int
5050 \bool_if:NT \l__enumext_show_answer_bool
5051 {
5052 __enumext_print_keyans_box:NN \l__enumext_labelwidth_i_dim \l__enumext_labelsep_i_dim
5053 }
5054 \bool_if:NT \l__enumext_show_position_bool
5055 {
5056 \tl_set:Ne \l__enumext_mark_answer_sym_tl
5057 {
5058 \group_begin:
5059 \exp_not:N \normalfont
5060 \exp_not:N \footnotesize [\int_eval:n
5061 {
5062 \prop_count:c { g__enumext_ \l__enumext_store_name_tl _prop }
5063 }
5064]
5065 \group_end:
5066 }
5067 __enumext_print_keyans_box:NN \l__enumext_labelwidth_i_dim \l__enumext_labelsep_i_dim
5068 }
5069 }

```

(End of definition for \\_\_enumext\_starred\_item\_viii:w and others.)

```

__enumext_keyans_wrapper_label_viii:n
__enumext_fake_make_label_viii:n

```

The implementation at this is very similar to that of the `enumext*` environment.

```

5070 \cs_new_protected:Npn __enumext_keyans_wrapper_label_viii:n #1
5071 {
5072 \bool_lazy_all:nT
5073 {
5074 { \bool_if_p:N \l__enumext_wrap_label_viii_bool }
5075 { \bool_if_p:N \l__enumext_show_answer_bool }
5076 { \int_compare_p:nNn { \l__enumext_item_star_exec_int } = { 1 } }
5077 { \cs_if_exist_p:N __enumext_keyans_wrapper_item:n }
5078 }
5079 {
5080 \cs_set_eq:NN __enumext_wrapper_label_viii:n __enumext_keyans_wrapper_item:n
5081 }
5082 \bool_if:NTF \l__enumext_wrap_label_viii_bool
5083 {
5084 __enumext_wrapper_label_viii:n {#1}
5085 }
5086 { #1 }
5087 }
5088 \cs_new_protected_nopar:Npn __enumext_fake_make_label_viii:n #1
5089 {
5090 \legacy_if:nT { @noitemarg }
5091 {
5092 \legacy_if_set_false:n { @noitemarg }
5093 \legacy_if:nT { @nmbrrlist }
5094 {
5095 \refstepcounter{enumXviii}
5096 }
5097 }
5098 \bool_if:NT \l__enumext_item_starred_viii_bool
5099 {
5100 __enumext_starred_item_exec:
5101 }
5102 \makebox[\l__enumext_labelwidth_viii_dim][\l__enumext_align_label_viii_str]
5103 {
5104 \tl_use:N \l__enumext_label_font_style_viii_tl
5105 __enumext_keyans_wrapper_label_viii:n {#1}
5106 }
5107 \skip_horizontal:N \l__enumext_labelsep_viii_dim \ignorespaces
5108 }

```

(End of definition for \\_\_enumext\_keyans\_wrapper\_label\_viii:n and \\_\_enumext\_fake\_make\_label\_viii:n.)

### 13.45.2 Real definition of \item in keyans\*

The implementation at this is very similar to that of the `enumext*` environment.

```

5109 \cs_new_protected_nopar:Npn __enumext_start_item_viii:w [#1]
5110 {
5111 \cs_set_eq:NN __enumext_stop_item_tmp_viii: __enumext_stop_item_viii:
5112 \hbox_set_to_wd:Nnw \l__enumext_item_text_viii_box
5113 {
5114 \l__enumext_joined_width_viii_dim
5115 + \l__enumext_labelwidth_viii_dim
5116 + \l__enumext_labelsep_viii_dim
5117 }
5118 __enumext_renew_footnote_starred:
5119 __enumext_start_list_tag:n {keyans*}
5120 __enumext_fake_make_label_viii:n {#1}
5121 __enumext_stop_start_list_tag:
5122 __enumext_minipage:w [t]{ \l__enumext_joined_width_viii_dim }
5123 \dim_set_eq:NN \parindent \l__enumext_listparindent_viii_dim
5124 \skip_set_eq:NN \parskip \l__enumext_parsep_viii_skip
5125 __enumext_unskip_unkern:
5126 __enumext_unskip_unkern:
5127 \skip_horizontal:n { -\l__enumext_listparindent_viii_dim } \ignorespaces
5128 \tl_use:N \l__enumext_fake_item_indent_viii_tl
5129 \bool_if:NT \l__enumext_item_starred_viii_bool
5130 {
5131 __enumext_keyans_show_item_opt:
5132 }
5133 \tl_use:N \l__enumext_after_list_args_viii_tl
5134 }
5135 \cs_new_protected_nopar:Nn __enumext_stop_item_viii:
5136 {
5137 __enumext_endminipage:
5138 __enumext_stop_list_tag:n {keyans*}
5139 \hbox_set_end:
5140 \int_set:Nn \hbadness { 10000 }
5141 \box_use_drop:N \l__enumext_item_text_viii_box
5142 \int_compare:nNnTF
5143 { \l__enumext_item_column_pos_viii_int } = { \l__enumext_columns_viii_int }
5144 {
5145 \par\noindent
5146 \int_zero:N \l__enumext_item_column_pos_viii_int
5147 }
5148 {
5149 \skip_horizontal:N \l__enumext_columns_sep_viii_dim
5150 }
5151 }

```

(End of definition for \\_\_enumext\_start\_item\_viii:w and \\_\_enumext\_stop\_item\_viii:.)

\\_\_enumext\_remove\_extra\_parsep\_viii:

The implementation at this is very similar to that of the `enumext*` environment.

```

5152 \cs_new_protected:Nn __enumext_remove_extra_parsep_viii:
5153 {
5154 \int_compare:nNnT
5155 {
5156 \int_mod:nn
5157 { \g__enumext_item_count_all_viii_int }
5158 { \l__enumext_columns_viii_int }
5159 }
5160 =
5161 { 0 }
5162 {
5163 \para_end:
5164 \skip_vertical:n { -\l__enumext_itemsep_viii_skip }
5165 \skip_vertical:N \c_zero_skip
5166 \int_gzero:N \g__enumext_item_count_all_viii_int
5167 }
5168 }

```

(End of definition for \\_\_enumext\_remove\_extra\_parsep\_viii:.)

### 13.46 The command \getkeyans

```
\getkeyans
__enumext_getkeyans_aux:n
__enumext_getkeyans:nn
```

The `\getkeyans` command takes a *mandatory argument* of the form  $\langle \text{store name} : \text{position} \rangle$ . Retrieve a “single content” stored by `\anskey`, `\anspic*` and `\item*` and `anskey*` from *prop list* defined by `save-ans` key.

```
5169 \NewDocumentCommand \getkeyans { m }
5170 {
5171 \exp_args:Ne __enumext_getkeyans_aux:n
5172 { \tl_to_str:e { \text_expand:n {#1} } }
5173 }
```

The internal function `__enumext_getkeyans_aux:n` is in charge of *splitting* the *mandatory argument* using “.”. If “.” is omitted it will return an error.

```
5174 \cs_new_protected:Npn __enumext_getkeyans_aux:n #1
5175 {
5176 \str_if_in:nnTF {#1} { : }
5177 {
5178 \use:e
5179 {
5180 \cs_set:Npn \exp_not:N __enumext_tmp:w ##1 \c_colon_str ##2 \scan_stop:
5181 { {##1} {##2} }
5182 }
5183 \exp_after:wN __enumext_getkeyans:nn __enumext_tmp:w #1 \scan_stop:
5184 }
5185 { \msg_error:nnn { enumext } { missing-colon } {#1} }
5186 }
```

The internal function `__enumext_getkeyans:nn` will check for the existence of the *prop list*, if it does not exist it will return an error message, then it will fetch the content specified by the *second argument* from *prop list*.

```
5187 \cs_new_protected:Npn __enumext_getkeyans:nn #1 #2
5188 {
5189 \prop_if_exist:cTF { g__enumext_#1_prop }
5190 {
5191 \prop_item:cn { g__enumext_#1_prop }{#2}
5192 }
5193 {
5194 \msg_error:nnn { enumext } { undefined-storage-anskey } {#1}
5195 }
5196 }
```

(End of definition for `\getkeyans`, `__enumext_getkeyans_aux:n`, and `__enumext_getkeyans:nn`. This function is documented on page 17.)

### 13.47 The command \printkeyans

The `\printkeyans` command prints “all stored content” in the *sequence* defined by the `save-ans` key.

The first thing we will do is define a set of  $\langle \text{filtered keys} \rangle$  with which we will control the options of the different nesting levels for the environment `enumext` and `enumext*` by storing their values in the list of tokens `\l__enumext_print_keyans_X_tl`.

The variable `\l__enumext_print_keyans_starred_tl` will have the default  $\langle \text{keys} \rangle$  for `\printkeyans*` and will be set by `\setenumext[⟨print*⟩]` and the variable `\l__enumext_print_keyans_vii_tl` will have the default keys for the environment `enumext*` nested within the *sequence* and will be set by `\setenumext[⟨print,*⟩]`, the rest of the variables will be for the environment `enumext` and will be set by `\setenumext[⟨print,level⟩]`.

```
5197 \keys_define:nn { enumext / print }
5198 {
5199 print* .code:n = \keys_precompile:neN { enumext / enumext* }
5200 { __enumext_filter_save_key:n {#1} }
5201 \l__enumext_print_keyans_starred_tl, % starred cmd
5202 print* .initial:n = { nosep, label=\arabic*, columns=2, first=\small, font=\small },
5203 print-1 .code:n = \keys_precompile:neN { enumext / level-1 }
5204 { __enumext_filter_save_key:n {#1} }
5205 \l__enumext_print_keyans_i_tl,
5206 print-1 .initial:n = { nosep, label=\arabic*, columns=2, first=\small, font=\small },
5207 print-2 .code:n = \keys_precompile:neN { enumext / level-2 }
5208 { __enumext_filter_save_key:n {#1} }
5209 \l__enumext_print_keyans_ii_tl,
5210 print-2 .initial:n = { nosep, label=(\alph*), first=\small, font=\small },
5211 print-3 .code:n = \keys_precompile:neN { enumext / level-3 }
5212 { __enumext_filter_save_key:n {#1} }
```

```

5213 \l__enumext_print_keyans_iii_tl,
5214 print-3 .initial:n = { nosep, label=\roman*., first=\small, font=\small },
5215 print-4 .code:n = \keys_precompile:neN { enumext / level-4 }
5216 { \l__enumext_filter_save_key:n {#1} }
5217 \l__enumext_print_keyans_iv_tl,
5218 print-4 .initial:n = { nosep, label=\Alph*., first=\small, font=\small },
5219 print-* .code:n = \keys_precompile:neN { enumext / enumext* }
5220 { \l__enumext_filter_save_key:n {#1} }
5221 \l__enumext_print_keyans_vii_tl, % starred nested
5222 print-* .initial:n = { nosep, label=\arabic*., first=\small, font=\small },
5223 }

```

- The reason for storing  $\langle keys \rangle$  in token lists using `\keys_precompile:neN` is because the keys are set via `\setenumext` but are later executed by running the command `\printkeyans` and they are not handled directly by its *optional argument*, except those related to the *first* opening level.

`\printkeyans`

`\l__enumext_printkeyans:nnn`

Create a user command to print “*all stored content*” in *sequence* for `\anskey`, `anskey*`, `\item*` and `\anspic*`. Within a group we will run our “*precompiled keys*” and then call the internal function `\l__enumext_printkeyans:nnn`.

```

5224 \NewDocumentCommand \printkeyans { s O{} m }
5225 {
5226 \group_begin:
5227 \tl_use:N \l__enumext_print_keyans_i_tl
5228 \tl_use:N \l__enumext_print_keyans_ii_tl
5229 \tl_use:N \l__enumext_print_keyans_iii_tl
5230 \tl_use:N \l__enumext_print_keyans_iv_tl
5231 \tl_use:N \l__enumext_print_keyans_vii_tl
5232 \l__enumext_printkeyans:nnn { #1 } { #2 } { #3 }
5233 \group_end:
5234 }

```

The internal function `\l__enumext_printkeyans:nnn` will check for the existence of the *sequence*, if it does not exist it will return an error message, then it will check if not empty.

```

5235 \cs_new_protected:Npn \l__enumext_printkeyans:nnn #1 #2 #3
5236 {
5237 \seq_if_exist:cTF { g__enumext_#3_seq }
5238 {
5239 \seq_if_empty:cF { g__enumext_#3_seq }
5240 {

```

If the *starred argument* `*` is present we will check that the environment `enumext*` is not saved in the *sequence*, then execute the variable `\l__enumext_print_keyans_starred_tl` that contains the default  $\langle keys \rangle$  for the environment `enumext*`, we set `\l__enumext_base_line_fix_bool` and `\l__enumext_print_keyans_star_bool` to true for *baseline correction*, open the `enumext*` environment passing the *optional argument* and map the *sequence*, then set `\l__enumext_base_line_fix_bool` and `\l__enumext_print_keyans_star_bool` to false.

```

5241 \bool_if:nTF {#1}
5242 {
5243 \seq_if_in:cnTF { g__enumext_#3_seq } { \end{enumext*} }
5244 {
5245 \msg_error:nnnn { enumext } { print-starred } {#3} { enumext* }
5246 }
5247 {
5248 \tl_use:N \l__enumext_print_keyans_starred_tl
5249 \bool_set_true:N \l__enumext_base_line_fix_bool
5250 \bool_set_true:N \l__enumext_print_keyans_star_bool
5251 \begin{enumext*}[#2]
5252 \seq_map_inline:cn { g__enumext_#3_seq } { ##1 }
5253 \end{enumext*}
5254 \bool_set_false:N \l__enumext_base_line_fix_bool
5255 \bool_set_false:N \l__enumext_print_keyans_star_bool
5256 }
5257 }

```

Otherwise it will open the environment `enumext` passing the *optional argument* to the “*first level*” then map the *sequence*.

```

5258 {
5259 \begin{enumext}[#2]
5260 \seq_map_inline:cn { g__enumext_#3_seq } { ##1 }
5261 \end{enumext}
5262 }
5263 }

```

```

5264 }
5265 {
5266 \msg_error:nnn { enumext } { undefined-storage-anskey } {#3}
5267 }
5268 }

```

(End of definition for `\printkeyans` and `\__enumext_printkeyans:nnn`. This function is documented on page 18.)

### 13.48 The command `\setenumext`

The command `\setenumext` will be in charge of managing the `⟨keys⟩` passed to all environments and to the `\printkeyans` command. We must take precautions with the `enumext*` environment and “first level” of the `enumext` environment so as not to capture `⟨keys⟩` that complicate us.

The function `\__enumext_filter_first_level:n` will be in charge of filtering the `⟨keys⟩` passed to the environment `enumext*` and “first level” of the environment `enumext`.

```

__enumext_filter_first_level:n
__enumext_filter_first_level_key:n
__enumext_filter_first_level_pair:nn
5269 \cs_new:Npn __enumext_filter_first_level:n #1
5270 {
5271 \use:e
5272 {
5273 \keyval_parse:NNn
5274 __enumext_filter_first_level_key:n
5275 __enumext_filter_first_level_pair:nn {#1}
5276 }
5277 }

```

The function `\__enumext_filter_first_level_key:n` will be responsible for filtering the `⟨keys⟩` that are passed “without value” by excluding the keys `resume` and `resume*`.

```

5278 \cs_new:Npn __enumext_filter_first_level_key:n #1
5279 {
5280 \str_case:nnF {#1}
5281 {
5282 { resume } {}
5283 { resume* } {}
5284 }
5285 { , { \exp_not:n {#1} } }
5286 }

```

The function `\__enumext_filter_first_level_pair:nn` will be responsible for filtering the `⟨keys⟩` that are passed “with value” by excluding the `series`, `resume` and `save-ans` keys.

```

5287 \cs_new:Npn __enumext_filter_first_level_pair:nn #1#2
5288 {
5289 \str_case:nnF {#1}
5290 {
5291 { series } {}
5292 { resume } {}
5293 { save-ans } {}
5294 }
5295 { , { \exp_not:n {#1} } = { \exp_not:n {#2} } }
5296 }

```

(End of definition for `\__enumext_filter_first_level:n`, `\__enumext_filter_first_level_key:n`, and `\__enumext_filter_first_level_pair:nn`.)

Now define a “meta families” of `⟨keys⟩` to access from `\setenumext`.

```

5297 \keys_define:nn { enumext / meta-families }
5298 {
5299 enumext-1 .code:n =
5300 {
5301 \keys_set:ne { enumext / level-1 }
5302 {
5303 __enumext_filter_first_level:n {#1}
5304 }
5305 } ,
5306 enumext-2 .code:n = { \keys_set:nn { enumext / level-2 } {#1} } ,
5307 enumext-3 .code:n = { \keys_set:nn { enumext / level-3 } {#1} } ,
5308 enumext-4 .code:n = { \keys_set:nn { enumext / level-4 } {#1} } ,
5309 keyans .code:n = { \keys_set:nn { enumext / keyans } {#1} } ,
5310 enumext* .code:n =
5311 {
5312 \keys_set:ne { enumext / enumext* }
5313 {
5314 __enumext_filter_first_level:n {#1}

```

```

5315 }
5316 },
5317 keyans* .code:n = { \keys_set:nn { enumext / keyans* } {#1} } ,
5318 print* .code:n = { \keys_set:nn { enumext / print } { print* = {#1} } } ,
5319 print-1 .code:n = { \keys_set:nn { enumext / print } { print-1 = {#1} } } ,
5320 print-2 .code:n = { \keys_set:nn { enumext / print } { print-2 = {#1} } } ,
5321 print-3 .code:n = { \keys_set:nn { enumext / print } { print-3 = {#1} } } ,
5322 print-4 .code:n = { \keys_set:nn { enumext / print } { print-4 = {#1} } } ,
5323 print-* .code:n = { \keys_set:nn { enumext / print } { print-* = {#1} } } ,
5324 unknown .code:n = { \msg_error:nn { enumext } { unknown-key-family } } ,
5325 }

```

We store them in the constant sequence `\c__enumext_all_families_seq` separated by commas.

```

5326 \seq_const_from_clist:Nn \c__enumext_all_families_seq
5327 {
5328 enumext-1, enumext-2, enumext-3, enumext-4, keyans, enumext*,
5329 keyans*, print-1, print-2, print-3, print-4, print-*, print*,
5330 }

```

`\setenumext` Now we define the user command `\setenumext`.

```

__enumext_set_parse:n 5331 \NewDocumentCommand \setenumext { 0{enumext,1} +m }
__enumext_set_error:nn 5332 {
5333 \seq_clear:N \l__enumext_setkey_tmpa_seq
5334 \seq_set_from_clist:Nn \l__enumext_setkey_tmpb_seq {#1}
5335 \int_set:Nn \l__enumext_setkey_tmpa_int
5336 {
5337 \seq_count:N \l__enumext_setkey_tmpb_seq
5338 }
5339 \int_compare:nNnTF { \l__enumext_setkey_tmpa_int } > { 1 }
5340 {
5341 \seq_pop_left:NN \l__enumext_setkey_tmpb_seq \l__enumext_setkey_tmpa_tl
5342 \seq_map_function:NN \l__enumext_setkey_tmpb_seq __enumext_set_parse:n
5343 \seq_set_map_e:NNn \l__enumext_setkey_tmpa_seq \l__enumext_setkey_tmpa_seq
5344 {
5345 \tl_use:N \l__enumext_setkey_tmpa_tl - ##1
5346 }
5347 }
5348 {
5349 \seq_put_right:Ne \l__enumext_setkey_tmpa_seq { \tl_trim_spaces:n {#1} }
5350 }
5351 \seq_if_empty:NNTF \l__enumext_setkey_tmpa_seq
5352 { \seq_map_inline:Nn \c__enumext_all_families_seq }
5353 { \seq_map_inline:Nn \l__enumext_setkey_tmpa_seq }
5354 {
5355 \keys_set:nn { enumext / meta-families } { ##1 = {#2} }
5356 }
5357 }

```

Internal functions used by the `\setenumext` command.

```

5358 \cs_new_protected:Npn __enumext_set_parse:n #1
5359 {
5360 \tl_set:Ne \l__enumext_setkey_tmpb_tl { \tl_trim_spaces:n {#1} }
5361 \clist_map_inline:nn { 0, 1, 2, 3, 4, * } % <- max level
5362 { \tl_remove_all:Nn \l__enumext_setkey_tmpb_tl {##1} }
5363 \tl_if_empty:NNTF \l__enumext_setkey_tmpb_tl
5364 {
5365 \seq_put_right:Ne \l__enumext_setkey_tmpa_seq
5366 { \tl_trim_spaces:n {#1} }
5367 }
5368 { __enumext_set_error:nn {#1} { } }
5369 }
5370 \cs_new_protected:Npn __enumext_set_error:nn #1 #2
5371 { \msg_error:nnn { enumext } { invalid-key } {#1} {#2} }

```

(End of definition for `\setenumext`, `\__enumext_set_parse:n`, and `\__enumext_set_error:nn`. This function is documented on page 6.)

### 13.49 The command `\setenumextmeta`

The command `\setenumextmeta` will be responsible for adding new “meta-keys” for the `enumext` and `enumext*` environments. The implementation code was given by Jonathan P. Spratte (@Skillmon) answer in [Add .meta key to existing keys \(l3keys\)](#).

```
\setenumextmeta
```

First we will create a prop list `\c__enumext_meta_paths_prop` to handle the *optional argument*.

```
\c__enumext_meta_paths_prop
__enumext_add_meta_key:nnn
__enumext_def_meta_key:nnn
__enumext_def_meta_key:Vnn
```

```
5372 \prop_const_from_keyval:Nn \c__enumext_meta_paths_prop
5373 {
5374 {enumext,1} = level-1,
5375 {enumext,2} = level-2,
5376 {enumext,3} = level-3,
5377 {enumext,4} = level-4,
5378 {enumext*} = enumext*
5379 }
```

Now we create the user command taking care that unknown cannot be passed as an argument.

```
5380 \NewDocumentCommand \setenumextmeta { s O{enumext,1} m +m }
5381 {
5382 \str_if_eq:eeTF { \tl_trim_spaces:n {#3} } { unknown }
5383 { \msg_error:nn { enumext } { prohibited-unknown } }
5384 {
5385 \bool_if:nTF {#1}
5386 {
5387 \int_step_inline:nn { 4 }
5388 { __enumext_add_meta_key:nnn { enumext, ##1 } {#3} {#4} }
5389 __enumext_add_meta_key:nnn { enumext* } {#3} {#4}
5390 }
5391 { __enumext_add_meta_key:nnn {#2} {#3} {#4} }
5392 }
5393 }
```

The internal functions `\__enumext_add_meta_key:nnn` and `\__enumext_def_meta_key:nnn` will check the *optional argument* and create the “meta-key”.

```
5394 \cs_new_protected:Npn __enumext_add_meta_key:nnn #1
5395 {
5396 \tl_set:Nn \l__enumext_meta_path_tl {#1}
5397 \tl_replace_all:Nnn \l__enumext_meta_path_tl { ~ } {}
5398 \prop_get:NVNTF
5399 \c__enumext_meta_paths_prop \l__enumext_meta_path_tl \l__enumext_meta_path_tl
5400 { __enumext_def_meta_key:Vnn \l__enumext_meta_path_tl }
5401 {
5402 \msg_error:nnn { enumext } { unknown-set } {#1}
5403 \use_none:nn
5404 }
5405 }
5406 \cs_new_protected:Npn __enumext_def_meta_key:nnn #1#2#3
5407 {
5408 \bool_lazy_or:nnTF
5409 { \keys_if_exist_p:nn { enumext / #1 } {#2} }
5410 { \keys_if_exist_p:nn { enumext / enumext* } {#2} }
5411 { \msg_error:nnn { enumext } { already-defined } {#2} }
5412 {
5413 \keys_define:nn { enumext / #1 }
5414 {
5415 #2 .meta:n = {#3},
5416 #2 .value_forbidden:n = true
5417 }
5418 }
5419 }
5420 \cs_generate_variant:Nn __enumext_def_meta_key:nnn { V }
```

(End of definition for `\setenumextmeta` and others. This function is documented on page 6.)

### 13.50 The command `\foreachkeyans`

The command `\foreachkeyans` will execute a *loop* over the *prop list* and return its contents. The implementation code is adapted from the answer provided by Enrico Gregorio (@egreg) in [Expand a .cs defined by key inside the function](#).

```
\foreachkeyans
```

We define a set of *⟨keys⟩* for command and we will save the default values of these in `\g__enumext_foreach_default_keys_tl` to avoid the use of group.

```
__enumext_parse_foreach_keys:nn
__enumext_parse_foreach_keys:nn
__enumext_foreach_keyans:nn
__enumext_foreach_add_body:n
```

```
5421 \keys_define:nn { enumext / foreach }
5422 {
5423 before .tl_set:N = \l__enumext_foreach_before_tl,
5424 before .value_required:n = true,
5425 after .tl_set:N = \l__enumext_foreach_after_tl,
5426 after .value_required:n = true,
```



```

5427 start .int_set:N = \l__enumext_foreach_start_int,
5428 start .value_required:n = true,
5429 stop .int_set:N = \l__enumext_foreach_stop_int,
5430 stop .value_required:n = true,
5431 step .int_set:N = \l__enumext_foreach_step_int,
5432 step .value_required:n = true,
5433 wrapper .cs_set_protected:Np = \l__enumext_foreach_wrapper:n #1,
5434 wrapper .value_required:n = true,
5435 sep .tl_set:N = \l__enumext_foreach_sep_tl,
5436 sep .value_required:n = true,
5437 unknown .code:n = { \l__enumext_parse_foreach_keys:n {#1} }
5438 }
5439 \keys_precompile:nnN { enumext / foreach }
5440 {
5441 before={},after={},start=1,step=1,stop=0,wrapper=#1,sep={; }
5442 }
5443 \g__enumext_foreach_default_keys_tl

```

Functions for handling unknown  $\langle keys \rangle$ .

```

5444 \cs_new_protected:Npn \l__enumext_parse_foreach_keys:nn #1#2
5445 {
5446 \tl_if_blank:nTF {#2}
5447 {
5448 \msg_error:nnn { enumext } { for-key-unknown } {#1}
5449 }
5450 {
5451 \msg_error:nnnn { enumext } { for-key-value-unknown } {#1} {#2}
5452 }
5453 }
5454 \cs_new_protected:Npn \l__enumext_parse_foreach_keys:n #1
5455 {
5456 \exp_args:NV \l__enumext_parse_foreach_keys:nn \l_keys_key_str {#1}
5457 }

```

We create the command.

```

5458 \NewDocumentCommand \foreachkeyans { +0{ } m }
5459 {
5460 \l__enumext_foreach_keyans:nn {#1} {#2}
5461 }

```

Finally the internal functions `\l__enumext_foreach_keyans:nn` and `\l__enumext_foreach_add_body:n` will loop through the prop list and print the contents.

```

5462 \cs_new_protected:Npn \l__enumext_foreach_keyans:nn #1 #2
5463 {
5464 \tl_use:N \g__enumext_foreach_default_keys_tl
5465 \keys_set:nn { enumext / foreach } {#1}
5466 \tl_set:Nn \l__enumext_foreach_name_prop_tl {#2}
5467 \prop_if_exist:cF { g__enumext_#2_prop }
5468 {
5469 \msg_error:nnn { enumext } { undefined-storage-anskey } {#2}
5470 }
5471 \int_compare:nNnT { \l__enumext_foreach_stop_int } = { 0 }
5472 {
5473 \int_set:Nn \l__enumext_foreach_stop_int
5474 { \prop_count:c { g__enumext_#2_prop } }
5475 }
5476 \seq_clear:N \l__enumext_foreach_print_seq
5477 \int_step_function:nnnN
5478 { \l__enumext_foreach_start_int }
5479 { \l__enumext_foreach_step_int }
5480 { \l__enumext_foreach_stop_int }
5481 \l__enumext_foreach_add_body:n
5482 \seq_use:NV \l__enumext_foreach_print_seq \l__enumext_foreach_sep_tl
5483 }
5484 \cs_new_protected:Npn \l__enumext_foreach_add_body:n #1
5485 {
5486 \seq_put_right:Ne \l__enumext_foreach_print_seq
5487 {
5488 \exp_not:V \l__enumext_foreach_before_tl
5489 \l__enumext_foreach_wrapper:n
5490 {
5491 \prop_item:cn { g__enumext_ \l__enumext_foreach_name_prop_tl _prop } {#1}

```

```

5492 }
5493 \exp_not:V \l__enumext_foreach_after_tl
5494 }
5495 }

```

(End of definition for `\foreachkeyans` and others. This function is documented on page 17.)

### 13.51 Messages

Message used by package-load for `multicol` and `hyperref` packages.

```

5496 \msg_new:nnn { enumext } { package-load }
5497 {
5498 The ~ '#1' ~ package ~ is ~ already ~ loaded.
5499 }
5500 \msg_new:nnn { enumext } { package-not-load }
5501 {
5502 The ~ '#1' ~ package ~ will ~ be ~ loaded ~ as ~ a ~ dependency.
5503 }
5504 \msg_new:nnn { enumext } { package-load-foot }
5505 {
5506 The ~ '#1' ~ package ~ is ~ loaded ~ with ~ the ~ option ~ '#2'.
5507 }

```

Message used in the creation of counters by `enumext` package.

```

5508 \msg_new:nnn { enumext } { counters }
5509 {
5510 The ~ counter ~ '#1' ~ is ~ already ~ defined ~ by ~ some ~ \\
5511 package ~ or ~ macro, ~ it ~ cannot ~ be ~ continued.
5512 }

```

Message used by `align` and `mark-pos` keys.

```

5513 \msg_new:nnn { enumext } { unknown-choice }
5514 {
5515 The ~ value ~ '#3' ~ for ~ '#1' ~ key ~ is ~ invalid ~ use ~ ('#2').
5516 }

```

Message used by reserved `anskey*` environment by `enumext` package.

```

5517 \msg_new:nnnn { enumext } { anskey-env-error }
5518 {
5519 The ~ '#1' ~ environment ~is~ reserved ~ by ~\\
5520 'enumext' ~ package, ~ It~ is~ already~ defined.
5521 }
5522 {
5523 The ~ anskey* ~ environment ~ is ~ defined ~ internally ~
5524 for ~ the ~ 'save-ans' ~ key.\\
5525 }

```

Message used in the creation of *prop list* by `enumext` package.

```

5526 \msg_new:nnn { enumext } { store-prop }
5527 {
5528 * ~ Package ~ enumext: ~ Creating ~
5529 \c_backslash_str g__enumext_#1_prop ~ \msg_line_context:.
5530 }
5531 \msg_new:nnn { enumext } { store-seq }
5532 {
5533 * ~ Package ~ enumext: ~ Creating ~
5534 \c_backslash_str g__enumext_#1_seq ~ \msg_line_context:.
5535 }
5536 \msg_new:nnn { enumext } { store-int }
5537 {
5538 * ~ Package ~ enumext: ~ Creating ~
5539 \c_backslash_str g__enumext_resume_#1_int ~ \msg_line_context:.
5540 }
5541 \msg_new:nnn { enumext } { prop-seq-int-hook }
5542 {
5543 * ~ Package ~ enumext: ~ Elements ~ in ~
5544 \c_backslash_str g__enumext_#1_prop ~ = ~ #2.\\
5545 * ~ Package ~ enumext: ~ Elements ~ in ~
5546 \c_backslash_str g__enumext_#1_seq ~ = ~ #3.\\
5547 * ~ Package ~ enumext: ~ Value ~ off ~
5548 \c_backslash_str g__enumext_resume_#1_int ~ = ~ #4.
5549 }
5550 \msg_new:nnn { enumext } { item-answer-hook }

```

```

5551 {
5552 * ~ Package ~ enumext: ~ Value ~ off ~
5553 \c_backslash_str g__enumext_item_number_int ~ = ~ #1.\\
5554 * ~ Package ~ enumext: ~ Value ~ off ~
5555 \c_backslash_str g__enumext_item_anskey_int ~ = ~ #2.\\
5556 * ~ Package ~ enumext: ~ Difference ~ item_number_int ~ - ~ item_anskey_int ~ = ~ #3.
5557 }

```

Message used by [*key = val*] system and `\setenumext` command.

```

5558 \msg_new:nnn { enumext } { invalid-key }
5559 {
5560 The ~ key ~ '#1' ~ is ~ not ~ know ~ the ~ level ~ #2.
5561 }
5562 \msg_new:nnn { enumext } { unknown-key-family }
5563 {
5564 Unknown~key~family~`\l_keys_key_str'~for~enumext.
5565 }

```

Messages used in length calculation.

```

5566 \msg_new:nnn { enumext } { width-negative }
5567 {
5568 Ignoring ~ negative ~ value ~ '#1=#2' ~ \msg_line_context:.\
5569 The ~ key ~ '#1'~ accepts ~ values ~ >= ~ 0pt.
5570 }
5571 \msg_new:nnn { enumext } { width-zero }
5572 {
5573 Invalid ~ '#1=#2' ~ \msg_line_context:.\
5574 The ~ key ~ '#1'~ accepts ~ values ~ > ~ 0pt.
5575 }

```

Messages used by `show-length` key in `enumext`.

```

5576 \msg_new:nnn { enumext } { list-lengths }
5577 {
5578 **** ~ Lengths ~ used ~ by ~ 'enumext' ~ level ~ '#2' ~ \msg_line_context:~\c_space_tl ****\\
5579 __enumext_show_length:nnn { dim } { labelsep } {#1}
5580 __enumext_show_length:nnn { dim } { labelwidth } {#1}
5581 __enumext_show_length:nnn { dim } { itemindent } {#1}
5582 __enumext_show_length:nnn { dim } { leftmargin } {#1}
5583 __enumext_show_length:nnn { dim } { rightmargin } {#1}
5584 __enumext_show_length:nnn { dim } { listparindent } {#1}
5585 __enumext_show_length:nnn { skip } { topsep } {#1}
5586 __enumext_show_length:nnn { skip } { parsep } {#1}
5587 __enumext_show_length:nnn { skip } { partopsep } {#1}
5588 __enumext_show_length:nnn { skip } { itemsep } {#1}
5589 ****
5590 }

```

Messages used by `show-length` key in `enumext*`, `keyans*` and `keyans`.

```

5591 \msg_new:nnn { enumext } { list-lengths-not-nested }
5592 {
5593 **** ~ Lengths ~ used ~ by ~ '#2' ~ environment ~ \msg_line_context:~\c_space_tl ****\\
5594 __enumext_show_length:nnn { dim } { labelsep } {#1}
5595 __enumext_show_length:nnn { dim } { labelwidth } {#1}
5596 __enumext_show_length:nnn { dim } { itemindent } {#1}
5597 __enumext_show_length:nnn { dim } { leftmargin } {#1}
5598 __enumext_show_length:nnn { dim } { rightmargin } {#1}
5599 __enumext_show_length:nnn { dim } { listparindent } {#1}
5600 __enumext_show_length:nnn { skip } { topsep } {#1}
5601 __enumext_show_length:nnn { skip } { parsep } {#1}
5602 __enumext_show_length:nnn { skip } { partopsep } {#1}
5603 __enumext_show_length:nnn { skip } { itemsep } {#1}
5604 ****
5605 }

```

Messages used by `ref` key.

```

5606 \msg_new:nnn { enumext } { key-ref-empty }
5607 {
5608 Key ~ 'ref' ~ need ~ a ~ value ~ in ~ '#1'~ \msg_line_context:.
5609 }

```

Messages used by `save-ans` key.

```

5610 \msg_new:nnn { enumext } { save-ans-empty }
5611 {
5612 Key ~ 'save-ans' ~ need ~ a ~ value ~ in ~ '#1'~ \msg_line_context:.

```

```

5613 }
5614 \msg_new:nnn { enumext } { save-ans-log }
5615 {
5616 * ~ Package ~ enumext: ~ Start ~ #1\c_space_tl with ~ save-ans=#2 ~ \msg_line_context:.
5617 }
5618 \msg_new:nnn { enumext } { save-ans-log-hook }
5619 {
5620 * ~ Package ~ enumext: ~ Stop ~ #1\c_space_tl with ~ save-ans=#2 ~ \msg_line_context:.
5621 }
5622 \msg_new:nnn { enumext } { save-ans-hook }
5623 {
5624 Stop ~ storing ~ for ~ 'save-ans=#1' ~ \msg_line_context:.
5625 }

```

Messages used by the internal system to check answer used by `check-ans` key.

```

5626 \msg_new:nnn { enumext } { need-save-ans }
5627 {
5628 Key ~ '#1'~ works ~ only ~ with ~ the ~ 'save-ans' ~ key ~ in ~ '#2'~ \msg_line_context:.
5629 }
5630 \msg_new:nnn { enumext } { items-same-answer }
5631 {
5632 *****\
5633 * ~ Package ~ enumext: ~ Checking ~ answers ~ in ~ '#1' ~
5634 for ~ \c_left_brace_str #2 \c_right_brace_str\
5635 * ~ started ~ #3 ~ and ~ close ~ \msg_line_context: : ~
5636 'OK', ~ all ~ items ~ with ~ answer.\
5637 *****
5638 }
5639 \msg_new:nnn { enumext } { item-greater-answer }
5640 {
5641 Checking ~ answers ~ in ~ '#1' ~ for ~ \c_left_brace_str #2 \c_right_brace_str\
5642 started ~ #3 ~ and ~ close ~ \msg_line_context: : ~'NOT ~ OK'\
5643 Items ~ > ~ Answers.
5644 }
5645 \msg_new:nnn { enumext } { item-less-answer }
5646 {
5647 Checking ~ answers ~ in ~ '#1' ~ for ~ \c_left_brace_str #2 \c_right_brace_str\
5648 started ~ #3 ~ and ~ close ~ \msg_line_context: : ~'NOT ~ OK'\
5649 Items ~ < ~ Answers.
5650 }

```

Messages used by the internal system to check for “starred” `\item*` and `\anspic*` commands.

```

5651 \msg_new:nnn { enumext } { missing-starred }
5652 {
5653 Missing ~ '\c_backslash_str #1*' ~ #2.
5654 }
5655 \msg_new:nnn { enumext } { many-starred }
5656 {
5657 Many ~ '\c_backslash_str #1*' ~ #2.
5658 }

```

Messages used by `\printkeyans*` command.

```

5659 \msg_new:nnn { enumext } { print-starred }
5660 {
5661 \c_backslash_str printkeyans*:~ The ~ sequence ~ '#1' ~ already ~ contains ~
5662 #2 ~ environment ~ \msg_line_context:.
5663 }

```

Message for the nesting depth of the environment `enumext`.

```

5664 \msg_new:nnn { enumext } { list-too-deep }
5665 {
5666 Too ~ deep ~ nesting ~ for ~ 'enumext' ~ \msg_line_context:~ \
5667 The ~ maximum ~ level ~ of ~ nesting ~ is ~ 4.
5668 }

```

Messages used by `\anskey`, `anskey*` and `\anspic` commands.

```

5669 \msg_new:nnn { enumext } { anskey-unnumber-item }
5670 {
5671 Can't ~ store ~ with ~ a ~ unnumbered ~ \c_backslash_str item ~ \msg_line_context:.
5672 }
5673 \msg_new:nnn { enumext } { anskey-already-stored }
5674 {
5675 Content ~ already ~ stored ~ for ~ this ~ \c_backslash_str item ~ \msg_line_context:.

```

```

5676 }
5677 \msg_new:nnn { enumext } { anskey-empty-arg }
5678 {
5679 Can't ~ store ~ empty ~ content ~ \msg_line_context:.
5680 }
5681 \msg_new:nnn { enumext } { anskey-wrong-place }
5682 {
5683 Wrong ~ place ~ for ~ command ~ '\c_backslash_str #1' ~ \msg_line_context:~ \\
5684 '\c_backslash_str #1' ~ works ~ in ~ the ~ environment ~ '#2'.
5685 }
5686 \msg_new:nnn { enumext } { anskey-nested }
5687 {
5688 The ~ command ~ \c_backslash_str anskey~ can't ~ be ~ nested ~ \msg_line_context:.
5689 }
5690 \msg_new:nnn { enumext } { anskey-math-mode }
5691 {
5692 #1 ~ can't ~ work ~ in ~ math ~ mode ~ \msg_line_context:.
5693 }
5694 \msg_new:nnn { enumext } { anskey-env-wrong }
5695 {
5696 The ~ environment ~ anskey* ~ cannot ~ use ~ in ~ '#1' ~ \msg_line_context:.
5697 }
5698 \msg_new:nnn { enumext } { anspic-wrong-place }
5699 {
5700 Wrong ~ place ~ for ~ command ~ '\c_backslash_str #1' ~ \msg_line_context:~ \\
5701 '\c_backslash_str #1' ~ works ~ in ~ the ~ environment ~ '#2'.
5702 }
5703 \msg_new:nnn { enumext } { command-wrong-place }
5704 {
5705 Wrong ~ place ~ for ~ command ~ '\c_backslash_str #1' ~ \msg_line_context:~ \\
5706 '\c_backslash_str #1' ~ works ~ outside ~ the ~ environment ~ '#2'.
5707 }
5708 \msg_new:nnnn { enumext } { anskey-env-key-unknown }
5709 {
5710 The ~ key ~ '#1' ~ is ~ unknown ~ by ~ environment~
5711 'anskey*' ~ and ~ is ~ being ~ ignored.
5712 }
5713 {
5714 The ~ environment ~ 'anskey*' ~ does ~ not ~ have ~ a ~ key ~ called ~'#1'.\\
5715 Check ~ that ~ you ~ have ~ spelled ~ the ~ key ~ name ~ correctly.
5716 }
5717 \msg_new:nnnn { enumext } { anskey-env-key-value-unknown }
5718 {
5719 The ~ key ~ '#1=#2' ~ is ~ unknown ~ by ~ environment ~
5720 'anskey*' ~ and ~ is ~ being ~ ignored.
5721 }
5722 {
5723 The ~ environment ~ 'anskey*' ~ does ~ not ~ have ~ a ~ key ~ called ~'#1'.\\
5724 Check ~ that ~ you ~ have ~ spelled ~ the ~ key ~ name ~ correctly.
5725 }
5726 \msg_new:nnnn { enumext } { anskey-cmd-key-unknown }
5727 { The ~ key ~ '#1'~ is ~ unknown ~ by ~ '\c_backslash_str anskey' ~ and ~ is ~ being ~ ignored.}
5728 {
5729 The ~ command ~'\c_backslash_str anskey' ~ does ~ not ~ have ~ a ~ key ~ called ~'#1'.\\
5730 Check ~ that ~ you ~ have ~ spelled ~ the ~ key ~ name ~ correctly.
5731 }
5732 \msg_new:nnnn { enumext } { anskey-cmd-key-value-unknown }
5733 { The ~ key ~ '#1=#2' ~ is ~ unknown ~ by ~ '\c_backslash_str anskey' ~ and ~ is ~ being ~ ignored.}
5734 {
5735 The ~ command ~ '\c_backslash_str anskey' ~ does ~ not ~ have ~ a ~ key ~ called ~'#1'.\\
5736 Check ~ that ~ you ~ have ~ spelled ~ the ~ key ~ name ~ correctly.
5737 }

```

Messages used by **keyans**, **keyans\*** and **keyanspic** environment.

```

5738 \msg_new:nnn { enumext } { keyans-nested }
5739 {
5740 The ~ environment ~ 'keyans' ~ can't ~ be ~ nested ~ \msg_line_context:.
5741 }
5742 \msg_new:nnn { enumext } { keyans-wrong-level }
5743 {
5744 Wrong ~ level ~ position ~ for ~ 'keyans' ~ \msg_line_context:~ \\
5745 The ~ environment ~ 'keyans' ~ can ~ only ~ be ~ in ~ the ~ first ~ level.

```

```

5746 }
5747 \msg_new:nnn { enumext } { wrong-place }
5748 {
5749 Wrong ~ place ~ for ~ '#1' ~ environment ~\msg_line_context:~ \\
5750 '#1' ~ is ~ only ~ found ~ with ~ '#2' ~ in ~ 'enumext.
5751 }
5752 \msg_new:nnn { enumext } { keyanspic-nested }
5753 {
5754 The ~ environment ~ 'keyanspic' ~ can't ~ be ~ nested~ \msg_line_context:~.
5755 }
5756 \msg_new:nnn { enumext } { keyanspic-wrong-level }
5757 {
5758 Wrong ~ level ~ position ~ for ~ 'keyanspic' ~ \msg_line_context:~ \\
5759 The ~ environment ~ 'keyans' ~ can ~ only ~ be ~ in ~ the ~ first ~ level.
5760 }
5761 \msg_new:nnn { enumext } { keyanspic-item-cmd }
5762 {
5763 Can't ~ use ~ \c_backslash_str item ~ in ~ keyanspic ~ \msg_line_context:.
5764 }
5765 \msg_new:nnnn { enumext } { keyans-unknown-key }
5766 {
5767 The ~ key ~ '#1' ~ is ~ unknown ~ by ~ environment~
5768 '\l__enumext_envir_name_tl' ~ and ~ is ~ being ~ ignored.
5769 }
5770 {
5771 The ~ environment ~ '\l__enumext_envir_name_tl' ~ does ~ not
5772 ~ have ~ a ~ key ~ called ~ '#1'.\\
5773 Check ~ that ~ you ~ have ~ spelled ~ the ~ key ~ name ~ correctly.
5774 }
5775 \msg_new:nnnn { enumext } { keyans-unknown-key-value }
5776 {
5777 The ~ key ~ '#1=#2' ~ is ~ unknown ~ by ~ environment ~
5778 '\l__enumext_envir_name_tl' ~ and ~ is ~ being ~ ignored.
5779 }
5780 {
5781 The ~ environment ~ '\l__enumext_envir_name_tl' ~ does ~ not
5782 ~ have ~ a ~ key ~ called ~ '#1'.\\
5783 Check ~ that ~ you ~ have ~ spelled ~ the ~ key ~ name ~ correctly.
5784 }

```

Message used by unknown  $\langle keys \rangle$  in `enumext*` environment.

```

5785 \msg_new:nnnn { enumext } { starred-unknown-key }
5786 {
5787 The ~ key ~ '#1' ~ is ~ unknown ~ by ~ environment~
5788 '\l__enumext_envir_name_tl' ~ and ~ is ~ being ~ ignored.
5789 }
5790 {
5791 The ~ environment ~ '\l__enumext_envir_name_tl' ~ does ~ not
5792 ~ have ~ a ~ key ~ called ~ '#1'.\\
5793 Check ~ that ~ you ~ have ~ spelled ~ the ~ key ~ name ~ correctly.
5794 }
5795 \msg_new:nnnn { enumext } { starred-unknown-key-value }
5796 {
5797 The ~ key ~ '#1=#2' ~ is ~ unknown ~ by ~ environment ~
5798 '\l__enumext_envir_name_tl' ~ and ~ is ~ being ~ ignored.
5799 }
5800 {
5801 The ~ environment ~ '\l__enumext_envir_name_tl' ~ does ~ not
5802 ~ have ~ a ~ key ~ called ~ '#1'.\\
5803 Check ~ that ~ you ~ have ~ spelled ~ the ~ key ~ name ~ correctly.
5804 }

```

Message used by unknown  $\langle keys \rangle$  in `enumext` environment.

```

5805 \msg_new:nnnn { enumext } { standar-unknown-key }
5806 {
5807 The ~ key ~ '#1' ~ is ~ unknown ~ by ~ environment ~ '\l__enumext_envir_name_tl' \c_space_tl
5808 ~ on ~ level ~ \int_use:N \l__enumext_level_int \c_space_tl and ~ is ~ being ~ ignored.
5809 }
5810 {
5811 The ~ environment ~ '\l__enumext_envir_name_tl' ~ does ~ not
5812 ~ have ~ a ~ key ~ called ~ '#1' ~ on ~ level ~ \int_use:N \l__enumext_level_int.\\
5813 Check ~ that ~ you ~ have ~ spelled ~ the ~ key ~ name ~ correctly.

```

```

5814 }
5815 \msg_new:nnnn { enumext } { standar-unknown-key-value }
5816 {
5817 The ~ key ~ '#1=#2' ~ is ~ unknown ~ by ~ environment ~ '\l__enumext_envir_name_tl' \c_space_
5818 ~ on ~ level ~ \int_use:N \l__enumext_level_int \c_space_tl and ~ is ~ being ~ ignored.
5819 }
5820 {
5821 The ~ environment ~ '\l__enumext_envir_name_tl' ~ does ~ not
5822 ~ have ~ a ~ key ~ called ~ '#1' ~ on ~ level ~ \int_use:N \l__enumext_level_int.\\
5823 Check ~ that ~ you ~ have ~ spelled ~ the ~ key ~ name ~ correctly.
5824 }

```

Message used by unknown *⟨keys⟩* in `\foreachkeyans`.

```

5825 \msg_new:nnnn { enumext } { for-key-unknown }
5826 { The~key~'#1'~is~unknown~by~'\c_backslash_str foreachkeyans'~and~is~being~ignored.}
5827 {
5828 The~command~'\c_backslash_str foreachkeyans'~does~not~have~a~key~called~'#1'.\\
5829 Check~that~you~have~spelled~the~key~name~correctly.
5830 }
5831 \msg_new:nnnn { enumext } { for-key-value-unknown }
5832 { The~key~'#1=#2'~is~unknown~by~'\c_backslash_str foreachkeyans'~and~is~being~ignored. }
5833 {
5834 The~command~'\c_backslash_str foreachkeyans'~does~not~have~a~key~called~'#1'.\\
5835 Check~that~you~have~spelled~the~key~name~correctly.
5836 }

```

Messages used by `\getkeyans` command.

```

5837 \msg_new:nnn { enumext } { undefined-storage-anskey }
5838 {
5839 Storage ~ named ~ '#1' ~ is ~ not ~ defined ~ \msg_line_context:.
5840 }

```

Messages used by `\miniright` command.

```

5841 \msg_new:nnn { enumext } { missing-miniright }
5842 {
5843 Missing ~ '\c_backslash_str miniright' ~ in ~ \msg_line_context:.\\
5844 The ~ key ~ 'mini-env' ~ need ~ '\c_backslash_str miniright'.
5845 }
5846 \msg_new:nnn { enumext } { wrong-miniright-place }
5847 {
5848 Wrong ~ place ~ for ~ '\c_backslash_str miniright' ~ \msg_line_context:~ \\
5849 Works ~ in ~ 'enumext' ~ and ~ 'keyans' ~ with ~ key ~ 'mini-env'.
5850 }
5851 \msg_new:nnn { enumext } { wrong-miniright-use }
5852 {
5853 Wrong ~ use ~ for ~ '\c_backslash_str miniright' ~ \msg_line_context:~ \\
5854 '\c_backslash_str miniright' ~ need ~ a ~ key ~ 'mini-env'.
5855 }
5856 \msg_new:nnn { enumext } { wrong-miniright-starred }
5857 {
5858 Can't ~ use ~ \c_backslash_str miniright ~ in ~ starred ~ environments ~ \msg_line_context:.
5859 }
5860 \msg_new:nnn { enumext } { many-miniright-used }
5861 {
5862 Can't ~ use ~ \c_backslash_str miniright ~ more ~ than ~ once ~ \msg_line_context:.
5863 }

```

Messages used by `\setenumextmeta` command.

```

5864 \msg_new:nnn { enumext } { unknown-set }
5865 {
5866 Argument ~ [#1] ~ is ~ unknown ~ by ~ \c_backslash_str setenumextmeta ~ \msg_line_context:.
5867 }
5868 \msg_new:nnn { enumext } { already-defined }
5869 {
5870 The ~ key ~ '#1' ~ is ~ already ~ defined ~ \msg_line_context:.
5871 }
5872 \msg_new:nnn { enumext } { prohibited-unknown }
5873 {
5874 The ~ name ~ 'unknown' ~ can't ~ be ~ chosen~ for ~ a ~ meta ~ key ~ \msg_line_context:.
5875 }

```

Messages used by `enumext*` and `keyans*` environments.

```

5876 \msg_new:nnn { enumext } { nested }

```



```

5877 {
5878 The ~ environment ~ \l__enumext_envir_name_tl \c_space_tl can't ~ be ~ nested ~ \msg_line_con
5879 }
5880 \msg_new:nnn { enumext } { nested-horizontal }
5881 {
5882 The ~ environment ~ \l__enumext_envir_name_tl \c_space_tl can't ~ be ~ nested ~ in ~ '#1' ~ '
5883 }
5884 \msg_new:nnn { enumext } { item-joined }
5885 {
5886 Items ~ joined ~ (#1) ~ > ~ #2 ~ columns ~\msg_line_context:.
5887 }
5888 \msg_new:nnn { enumext } { item-joined-columns }
5889 {
5890 Not ~ space ~ to ~ join ~ items ~ (#1) ~ > ~ #2 ~\msg_line_context:.
5891 }

```

## 13.52 Finish package

Finish package implementation.

```

5892 \file_input_stop:
5893 \</package>

```

## 14 Index of Implementation

The *italic* numbers denote the pages where the corresponding entry is described, the numbers underlined and all others indicate the line on which they are implemented in the package code.

| Symbols                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                                 |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------|
| <code>\*</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | 229                             |
| <code>\+</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | 221                             |
| <code>\-</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | 221                             |
| <code>\\</code> 237, 2910, 4308, 4311, 5510, 5519, 5524, 5544, 5546, 5553, 5555, 5568, 5573, 5578, 5593, 5632, 5634, 5636, 5641, 5642, 5647, 5648, 5666, 5683, 5700, 5705, 5714, 5723, 5729, 5735, 5744, 5749, 5758, 5772, 5782, 5792, 5802, 5812, 5822, 5828, 5834, 5843, 5848, 5853                                                                                                                                                                                                                                                                                                                                                           |                                 |
| A                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                                 |
| <code>above</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | <u>1716</u>                     |
| <code>above*</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | <u>1716</u>                     |
| <code>\addvspace</code> 1283, 1311, 1354, 1357, 1525, 1528, 1625, 1631, 1669, 1675, 1696, 1702, 3753, 3914, 3932, 4192, 4196, 4556, 4571, 4617, 4631                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                                 |
| <code>after</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | <u>1113</u>                     |
| <code>align</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | <u>666</u>                      |
| <code>\Alph</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 41, 45, <u>46</u>               |
| <code>\Alpha</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | 608, 736, 780, 846, 5218        |
| <code>\alph</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 41, 45, <u>46</u>               |
| <code>\alpha</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | 609, 734, 5210                  |
| <code>\anskey</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 13, 79, 81, <u>2728</u>         |
| <code>anskey*</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 13, <u>2838</u>                 |
| <code>\anspic</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 16, 108, 111, <u>4206</u>       |
| <code>\anspic*</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | 73                              |
| <code>\arabic</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 33, <u>41</u>                   |
| <code>\arabic</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 607, 733, 779, 5202, 5206, 5222 |
| B                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                                 |
| <code>base-fix</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | <u>975</u>                      |
| <code>\baselineskip</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | <u>54</u>                       |
| <code>\baselineskip</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | 991, 998                        |
| <code>before</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | <u>1113</u>                     |
| <code>before*</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | <u>1113</u>                     |
| <code>below</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | <u>1716</u>                     |
| <code>below*</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | <u>1716</u>                     |
| bool commands:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                                 |
| <code>\bool_gset_false:N</code> 350, 351, 352, 3014, 3016, 4573, 4577, 4633                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                                 |
| <code>\bool_gset_true:N</code> 258, 268, 1216, 2209, 2215, 4542, 4574, 4606, 4634                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                                 |
| <code>\bool_if:NTF</code> . 401, 411, 428, 502, 509, 518, 525, 539, 552, 1738, 1752, 1765, 1776, 1787, 1798, 1809, 1820, 1869, 1886, 1891, 1899, 1926, 1964, 1969, 1976, 1980, 2002, 2007, 2015, 2022, 2053, 2061, 2154, 2360, 2370, 2449, 2473, 2480, 2504, 2602, 2624, 2664, 2678, 2682, 2732, 2751, 2775, 2829, 2840, 2929, 2966, 3030, 3065, 3080, 3155, 3166, 3170, 3189, 3202, 3244, 3278, 3314, 3329, 3350, 3489, 3510, 3594, 3604, 3637, 3642, 3708, 3734, 3784, 3842, 3897, 3922, 4126, 4190, 4208, 4227, 4279, 4306, 4535, 4551, 4557, 4600, 4614, 4618, 4707, 4717, 4805, 4811, 4818, 4834, 4927, 4937, 5050, 5054, 5082, 5098, 5129 |                                 |
| <code>\bool_if:nTF</code> 1676, 1703, 3300, 3468, 3552, 4248, 5241, 5385                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                                 |
| <code>\bool_if_p:N</code> 277, 292, 985, 986, 994, 995, 1648, 2033, 2034, 2042, 2043, 2167, 2193, 2206, 2207, 2212, 2213, 2537, 2547, 2559, 2574, 2575, 2609, 2650, 2651, 2952, 3142, 3143, 3180, 3181, 3502, 3503, 3681, 3683, 3694, 4257, 4258, 5074, 5075                                                                                                                                                                                                                                                                                                                                                                                    |                                 |
| <code>\bool_lazy_all:nTF</code> 275, 290, 983, 2165, 2191, 2535, 2544, 2557, 2572, 3500, 3679, 3692, 5072                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |                                 |
| <code>\bool_lazy_and:nnTF</code> 254, 264, 993, 1640, 1647, 2032, 2041, 2205, 2211, 2608, 2615, 2649, 2793, 2805, 2951, 2957, 3141                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |                                 |
| <code>\bool_lazy_or:nnTF</code> . . 2094, 2101, 3179, 4256, 5408                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                                 |
| <code>\bool_new:N</code> 30, 31, 32, 33, 34, 35, 36, 37, 60, 69, 93, 98, 99, 104, 105, 108, 127, 134, 135, 142, 149, 150, 153, 157, 159, 160, 177, 189, 191                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                                 |
| <code>\bool_not_p:n</code> 255, 265, 987, 1649, 2546, 2610, 2616, 2953, 2958, 3682, 3695                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                                 |
| <code>\bool_set_eq:NN</code> . . . . . 3253, 3447, 4758, 5002                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                                 |
| <code>\bool_set_false:N</code> 408, 1005, 2139, 2140, 2172, 2177, 2181, 2185, 2198, 2893, 3656, 3801, 3850, 3937, 4079, 4123, 4676, 4702, 4755, 4943, 4998, 5254, 5255                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                                 |
| <code>\bool_set_true:N</code> . 282, 283, 297, 298, 393, 396, 659, 1020, 1722, 1727, 1989, 2111, 2112, 2392, 2400, 2894, 3247, 3249, 3281, 3283, 3443, 3455, 3617, 3655, 3688, 3701, 3774, 3847, 3874, 4076, 4251, 4524, 4589, 4675, 4762, 4769, 4770, 4814, 4941, 5006, 5013, 5015, 5249, 5250                                                                                                                                                                                                                                                                                                                                                 |                                 |
| box commands:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                                 |
| <code>\box_dp:N</code> . . 1571, 1572, 1575, 1582, 1595, 1603, 1609, 1617, 4137, 4142, 4192, 4290                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                                 |
| <code>\box_ht:N</code> . . 1354, 1357, 1368, 1369, 1380, 1382, 1397, 1400, 1408, 1409, 1420, 1422, 1437, 1440, 1447, 1448, 1459, 1461, 1476, 1479, 1525, 1528, 1536, 1537, 1545, 1546, 1558, 1560                                                                                                                                                                                                                                                                                                                                                                                                                                               |                                 |
| <code>\box_ht_plus_dp:N</code> . . . . . 4132, 4200, 4236                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |                                 |
| <code>\box_new:N</code> . . . . . 66, 145, 146, 184, 190                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                                 |
| <code>\box_use_drop:N</code> . . . . . 4568, 4629, 4870, 5141                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                                 |
| <code>\box_wd:N</code> . . . . . 615                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                                 |
| C                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                                 |
| <code>\c</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | 229, 230, 882, 884, 896, 898    |
| <code>\catcode</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | 2910                            |
| <code>\cB</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | 230                             |
| <code>\cE</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | 230                             |
| <code>\centering</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 1678, 1705, 4334, 4561, 4622    |
| <code>check-ans</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | <u>2131</u>                     |
| Document class:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                                 |
| <code>article</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 47                              |
| clist commands:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                                 |
| <code>\clist_const:Nn</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | 196                             |
| <code>\clist_map_function:nN</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | 4317                            |
| <code>\clist_map_inline:Nn</code> . 665, 930, 1112, 1127, 1208, 1732                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                                 |
| <code>\clist_map_inline:nn</code> . 45, 56, 74, 82, 95, 107, 137, 168, 195, 643, 696, 716, 1025, 1046, 1222, 1838, 2078, 2145, 2333, 2357, 2389, 2532, 3074, 3372, 3384, 3424, 3581, 3584, 3612, 3624, 3627, 3647, 5361                                                                                                                                                                                                                                                                                                                                                                                                                         |                                 |
| <code>\columnbreak</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | 80                              |
| <code>\columnbreak</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | 2612                            |
| <code>columns</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | <u>1192</u>                     |
| <code>columns-sep</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | <u>1192</u>                     |
| <code>\columnsep</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 102                             |
| <code>\columnsep</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 3729, 3895                      |
| <code>\columnseprule</code> .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | 102                             |

\columnseprule . . . . . 3732, 3896

Commands provide by **enumext**:

\anskey . 30, 69, 70, 75, 76, 78–82, 88, 90, 101, 120, 130, 131, 138  
 \anspic\* . . . . . 30, 31, 73, 76, 88, 89, 111, 130, 131  
 \anspic . . . . . 30, 76, 108, 111, 138  
 \foreachkeyans . . . . . 134, 141  
 \getkeyans . . . . . 76, 130, 141  
 \item\* . 30, 31, 73, 76, 88, 89, 92, 96, 122, 127, 130, 131  
 \item . . . . . 92, 96, 115, 121, 123, 126, 127  
 \miniright . . . . . 29, 52, 60, 61, 103, 141  
 \printkeyans\* . . . . . 130  
 \printkeyans . . . . . 30, 76, 130, 131  
 \setenumextmeta . . . . . 133, 141  
 \setenumext . . . . . 30, 131–133, 137

Counters defined by **enumext**:

enumXiii . . . . . 28, 40  
 enumXii . . . . . 28, 40  
 enumXiv . . . . . 28, 40  
 enumXi . . . . . 28, 40  
 enumXviii . . . . . 28, 40  
 enumXvii . . . . . 28, 40, 122  
 enumXvi . . . . . 28, 40  
 enumXv . . . . . 28, 40

cs commands:

\cs\_generate\_variant:Nn . 201, 202, 617, 633, 888, 904, 2441, 2446, 2522, 2846, 3571, 4319, 5420  
 \cs\_if\_exist:NTF . . . . . 587  
 \cs\_if\_exist\_p:N . . . . . 3505, 5077  
 \cs\_if\_free:NTF . . . . . 2797, 2809  
 \cs\_new:Nn . . . . . 215  
 \cs\_new:Npn . 233, 1839, 1848, 1856, 2404, 2413, 2421, 5269, 5278, 5287  
 \cs\_new\_eq:NN . 377, 378, 383, 384, 413, 414, 417, 418  
 \cs\_new\_protected:Nn . 225, 239, 247, 273, 306, 336, 342, 348, 354, 360, 368, 388, 436, 440, 458, 470, 488, 500, 516, 532, 545, 566, 756, 817, 868, 981, 1128, 1132, 1136, 1140, 1144, 1148, 1152, 1156, 1160, 1164, 1168, 1172, 1176, 1180, 1184, 1188, 1223, 1235, 1268, 1285, 1296, 1313, 1339, 1360, 1485, 1511, 1531, 1564, 1586, 1621, 1627, 1733, 1747, 1761, 1772, 1783, 1794, 1805, 1816, 1897, 2000, 2013, 2030, 2051, 2079, 2084, 2109, 2150, 2160, 2203, 2218, 2225, 2234, 2239, 2244, 2249, 2258, 2263, 2268, 2447, 2471, 2478, 2502, 2509, 2523, 2749, 2768, 2784, 2847, 2883, 2914, 2949, 2991, 3012, 3020, 3063, 3078, 3106, 3139, 3175, 3187, 3200, 3286, 3296, 3307, 3323, 3339, 3464, 3482, 3516, 3526, 3648, 3677, 3706, 3713, 3743, 3760, 3782, 3804, 3840, 3864, 3881, 3906, 3920, 3941, 4098, 4301, 4315, 4320, 4344, 4354, 4385, 4514, 4533, 4579, 4598, 4662, 4689, 4696, 4705, 4715, 4740, 4881, 4925, 4956, 4962, 4983, 5042, 5152  
 \cs\_new\_protected:Npn 203, 207, 211, 421, 585, 602, 612, 618, 737, 781, 851, 875, 889, 1660, 1689, 1865, 1884, 1954, 1987, 2089, 2273, 2358, 2368, 2390, 2398, 2433, 2442, 2598, 2661, 2676, 2714, 2718, 2838, 2869, 2873, 2904, 3040, 3116, 3160, 3240, 3259, 3385, 3389, 3403, 3407, 3425, 3429, 3439, 3451, 3498, 3540, 3574, 3615, 3659, 3860, 4107, 4114, 4121, 4225, 4244, 4275, 4416, 4465, 4679, 4746, 4753, 4767, 4775, 4780, 4790, 4949, 4989, 4996, 5011, 5020, 5037, 5070, 5174, 5187, 5235, 5358, 5370, 5394, 5406, 5444, 5454, 5462, 5484  
 \cs\_new\_protected\_nopar:Nn . . . 4004, 4048, 4056, 4064, 4725, 4733, 4864, 4968, 4976, 5135  
 \cs\_new\_protected\_nopar:Npn . . 3996, 4012, 4796,

4842, 5088, 5109

\cs\_set:Npn . . . . . 2533, 2570, 5180  
 \cs\_set\_eq:NN . . 3508, 4652, 4653, 4844, 4914, 4915, 5080, 5111  
 \cs\_set\_protected:Nn . . . . 1051, 1067, 1080, 1092  
 \cs\_set\_protected:Npn . 41, 50, 67, 75, 90, 96, 130, 164, 175, 634, 644, 666, 701, 717, 763, 905, 931, 1007, 1030, 1104, 1113, 1192, 1209, 1716, 1827, 2070, 2131, 2290, 2334, 2376, 2525, 3067, 3361, 3377, 3417, 3572, 3613  
 \cs\_to\_str:N . . . . . 604, 627  
 \cs\_undefine:N . . . . . 2786, 2787, 2788, 2789

## D

\d . . . . . 221  
 \DeclareDocumentEnvironment . . . . . 570

dim commands:

\dim\_abs:n . . . . . 3545, 3550  
 \dim\_add:Nn . . . . . 4141, 4379, 4410  
 \dim\_compare:nNnTF . . 1053, 1069, 1082, 1094, 1372, 1384, 1412, 1424, 1451, 1463, 1540, 1548, 1662, 1691, 3542, 3547, 3553, 3559, 3561, 3563, 3718, 3765, 3868, 3885, 4116, 4356, 4372, 4387, 4403, 4516, 4581  
 \dim\_compare:nTF . . . . . 2634, 2979, 3807, 3944  
 \dim\_eval:n . . . . . 991, 4198, 4286  
 \dim\_gset\_eq:NN . . . . . 4525, 4590  
 \dim\_gzero:N . . . . . 3018, 4576, 4636  
 \dim\_new:N . 63, 70, 71, 72, 92, 139, 147, 148, 183, 185, 186, 192  
 \dim\_set:Nn . 615, 1021, 3276, 3545, 3550, 3552, 3555, 3556, 3560, 3562, 3565, 3566, 3568, 3721, 3768, 3806, 3870, 3887, 3943, 4130, 4234, 4322, 4358, 4365, 4389, 4396, 4451, 4500, 4518, 4583, 4792  
 \dim\_set\_eq:NN 724, 770, 839, 843, 3191, 3192, 3204, 3205, 3271, 3583, 3626, 3729, 3895, 4458, 4461, 4462, 4507, 4510, 4511, 4785, 4856, 5123  
 \dim\_sub:Nn . . . . . 3812, 3949, 4374, 4405  
 \dim\_use:N . 1054, 1062, 1663, 1673, 2512, 2515, 2520, 3291, 3293, 3346, 3719, 3723, 3724, 3726, 3766, 3771, 3772, 3778, 3809, 3814  
 \dim\_zero:N . . . . . 3618, 3732, 3896, 4143  
 \dim\_zero\_new:N . . . . . 584  
 \c\_zero\_dim 1056, 1070, 1083, 1095, 1663, 1691, 2636, 2981, 3542, 3547, 3553, 3560, 3719, 3766, 3809, 3868, 3885, 3946, 4116, 4356, 4372, 4387, 4403, 4516, 4581  
 \dimeval . . . . . 2297

## E

\end . . . 2475, 2506, 3750, 3911, 4180, 4336, 5243, 5253, 5261  
 end internal commands:

\end\_\_enumext\_mini\_page . 1671, 1698, 3793, 3931, 4540, 4604, 4630  
 \endgroup . . . . . 2910  
 \endlist . . . . . 378  
 \endminipage . . . . . 384  
 enumext . . . . . 5, 3818

enumext internal commands:

\l\_\_enumext\_\_ref\_the\_count\_tl . . . . . 43  
 \l\_\_enumext\_\_resume\_name\_tl . . . . . 65  
 \\_\_enumext\_add\_meta\_key:nnn . . 134, 5372, 5388, 5389, 5391, 5394  
 \\_\_enumext\_add\_pre\_parsep: . 53, 1233, 1235, 1235  
 \\_\_enumext\_after\_args\_exec: 51, 1128, 1140, 3831  
 \\_\_enumext\_after\_args\_exec\_v: 1144, 1156, 3964  
 \\_\_enumext\_after\_args\_exec\_vii: . . 1160, 1184  
 \\_\_enumext\_after\_args\_exec\_viii: . . . . . 1188

`\__enumext_after_env:nn` 85, 86, 88, 104, 117, 124, 207, 207, 558, 562, 2924, 3836, 4549, 4612, 4897  
`\__enumext_after_hyperref:` ... 36, 386, 386, 388  
`\l__enumext_after_list_args_v_tl` ..... 1158  
`\l__enumext_after_list_args_vii_tl` 1186, 4862  
`\l__enumext_after_list_args_viii_tl` .. 1190, 5133  
`\__enumext_after_list_vii:` 117, 120, 4660, 4696, 4696  
`\__enumext_after_list_viii:` ... 126, 4923, 4962, 4962  
`\__enumext_after_stop_list:` 51, 103, 1128, 1136, 3798  
`\__enumext_after_stop_list_v:` 1144, 1152, 3938  
`\l__enumext_after_stop_list_v_tl` ..... 1154  
`\__enumext_after_stop_list_vii:` .. 120, 1160, 1176, 4699  
`\l__enumext_after_stop_list_vii_tl` ... 1178  
`\__enumext_after_stop_list_viii:` . 1180, 4965  
`\l__enumext_after_stop_list_viii_tl` ... 1182  
`\l__enumext_align_label_pos_v_str` .... 3532  
`\l__enumext_align_label_pos_X_str` ..... 75  
`\l__enumext_align_label_vii_str` ..... 4831  
`\l__enumext_align_label_viii_str` ..... 5102  
`\l__enumext_align_label_X_str` ..... 175  
`\c__enumext_all_envs_clist` . 196, 665, 930, 1112, 1127, 1208, 1732  
`\c__enumext_all_families_seq` .. 133, 5326, 5352  
`\l__enumext_anskey_env_bool` 34, 84, 30, 283, 298, 2840  
`\__enumext_anskey_env_clean_vars:` . 87, 2945, 2949, 3012  
`\__enumext_anskey_env_define_keys:` 84, 2838, 2847, 2918  
`\__enumext_anskey_env_exec:` 85, 2843, 2914, 2914  
`\__enumext_anskey_env_make:n` 69, 84, 2114, 2838, 2838, 2846  
`\__enumext_anskey_env_reset_keys:` 85, 86, 2838, 2883, 2946  
`\__enumext_anskey_env_save_keys:` .. 86, 2926, 2949, 2949  
`\__enumext_anskey_env_store:` .. 87, 2942, 2949, 2991  
`\__enumext_anskey_env_unknown:n` 85, 2838, 2866, 2869  
`\__enumext_anskey_env_unknown:nn` . 2838, 2871, 2873  
`\l__enumext_anskey_level_int` .. 24, 2770, 2771  
`\__enumext_anskey_safe_inner:` . 83, 2743, 2749, 2768  
`\__enumext_anskey_safe_inner:n` ..... 82  
`\__enumext_anskey_safe_outer:` . 82, 2730, 2749, 2749  
`\__enumext_anskey_show_wrap_arg:n` . 80, 2661, 2661, 2680, 2695  
`\__enumext_anskey_show_wrap_left:n` 81, 2606, 2676, 2676  
`\__enumext_anskey_unknown:n` 82, 2698, 2712, 2714  
`\__enumext_anskey_unknown:nn` . 2698, 2716, 2718  
`\__enumext_anskey_wrapper:n` ..... 2294, 2674  
`\l__enumext_anspic_above_int` . 138, 4323, 4324, 4326  
`\__enumext_anspic_args:nnn` 111, 113, 4206, 4222, 4301  
`\l__enumext_anspic_args_seq` 111, 113, 138, 4220, 4335, 4348  
`\l__enumext_anspic_below_int` . 138, 4323, 4324, 4327  
`\l__enumext_anspic_body_box` ... 138, 4233, 4236  
`\__enumext_anspic_body_dim:n` .. 111, 4206, 4225, 4278  
`\l__enumext_anspic_body_htdp_dim` .. 111, 138, 4234, 4289  
`\__enumext_anspic_exec:` ..... 4206  
`\__enumext_anspic_exec:` ... 110, 113, 4175, 4344  
`\__enumext_anspic_label:nn` 112, 4206, 4244, 4281, 4296  
`\l__enumext_anspic_label_above_bool` ... 138, 4076, 4079, 4126, 4190, 4227, 4279, 4306  
`\l__enumext_anspic_label_box` .. 138, 4129, 4132  
`\l__enumext_anspic_label_htdp_dim` . 110, 138, 4130, 4136, 4200, 4288  
`\__enumext_anspic_label_pos:nnn` .. 112, 4206, 4275, 4304  
`\l__enumext_anspic_label_sep_skip` 4086, 4138, 4201, 4291, 4308  
`\l__enumext_anspic_layout_style_tl` 4088, 4346, 4351  
`\l__enumext_anspic_mini_pos_str` .. 138, 4077, 4080, 4333  
`\l__enumext_anspic_mini_width_dim` 138, 4246, 4322, 4333  
`\__enumext_anspic_print:n` 113, 4206, 4315, 4319, 4348, 4351  
`\__enumext_anspic_row:n` .. 113, 4206, 4317, 4320  
`\__enumext_anspic_start_list_tag:` 4020, 4048, 4303  
`\__enumext_anspic_stop_list_tag:` . 4020, 4064, 4313  
`\__enumext_anspic_stop_start_list_tag:` 4020, 4056, 4305  
`\__enumext_at_begin_document:n` .. 36, 203, 203, 375, 381  
`\l__enumext_base_line_fix_bool` 48, 131, 977, 986, 1005, 5249, 5254  
`\__enumext_before_args_exec:` 50, 102, 120, 1128, 1128, 3763  
`\__enumext_before_args_exec_v:` 1144, 1144, 3867  
`\__enumext_before_args_exec_vii:` . 1160, 1160, 4693  
`\__enumext_before_args_exec_viii:` 1164, 4959  
`\__enumext_before_env:nn` 84, 207, 211, 2791, 2803, 2815, 2916  
`\__enumext_before_keys_exec:` .. 50, 1128, 1132, 3828  
`\__enumext_before_keys_exec_v:` 1144, 1148, 3961  
`\__enumext_before_keys_exec_vii` ..... 1160  
`\__enumext_before_keys_exec_vii:` . 1168, 4647  
`\__enumext_before_keys_exec_viii:` 1172, 4909  
`\__enumext_before_list:` .. 102, 3760, 3760, 3822  
`\__enumext_before_list_v:` ... 3864, 3864, 3956  
`\__enumext_before_list_vii:` ... 120, 4642, 4689, 4689  
`\__enumext_before_list_viii:` .. 125, 4905, 4956, 4956  
`\l__enumext_before_no_starred_key_v_tl` 1150  
`\l__enumext_before_no_starred_key_vii_-tl` ..... 1170

```

\l__enumext_before_no_starred_key_viii_-
 tl 1174
\l__enumext_before_starred_key_v_tl ... 1146
\l__enumext_before_starred_key_vii_tl . 1162
\l__enumext_before_starred_key_viii_tl 1166
__enumext_calc_hspace:NNNNNN 98, 3540, 3540,
 3571, 3576, 3619
__enumext_check_ans_active: 71, 102, 120, 2150,
 2150, 3764, 4692
\g__enumext_check_ans_item_tl 90
\g__enumext_check_ans_key_bool .. 72, 149, 350,
 2209, 2215, 3030
\l__enumext_check_ans_key_bool 72, 2135, 2140,
 2206, 2212
__enumext_check_ans_key_hook: .. 72, 103, 120,
 2203, 2203, 3799, 4700
__enumext_check_ans_level: 71, 2150, 2156, 2160
__enumext_check_ans_log: 72, 87, 2249, 2249, 3034
__enumext_check_ans_log_msg_greater: 2249,
 2255, 2268
__enumext_check_ans_log_msg_less: 2249, 2253,
 2258
__enumext_check_ans_log_msg_same_ok: 2249,
 2254, 2263
__enumext_check_ans_msg_greater: 2225, 2231,
 2244
__enumext_check_ans_msg_less: 2225, 2229, 2234
__enumext_check_ans_msg_same_ok: 2225, 2230,
 2239
__enumext_check_ans_show: .. 72, 87, 2225, 2225,
 3032
\l__enumext_check_answers_bool . 69, 71, 82, 92,
 149, 2112, 2139, 2154, 2449, 2473, 2480, 2504, 2732,
 2929, 3155, 3244, 3278, 4811
__enumext_check_starred_cmd:n 34, 73, 90, 124,
 2273, 2273, 3967, 4188, 4922
\g__enumext_check_starred_cmd_int .. 96, 149,
 2276, 2282, 2287, 3462, 4255, 5049
\l__enumext_check_start_line_env_tl . 34, 149,
 313, 321, 329, 2279, 2285, 2288
\l__enumext_columns_sep_v_dim 3885, 3887, 3895
\l__enumext_columns_sep_vii_dim .. 4356, 4358,
 4367, 4379, 4455, 4878
\l__enumext_columns_sep_viii_dim . 4387, 4389,
 4398, 4410, 4504, 5149
\l__enumext_columns_v_int 1505, 1523, 1694, 3883,
 3891, 3903, 3908
\l__enumext_columns_vii_int .. 4361, 4364, 4368,
 4377, 4419, 4423, 4426, 4432, 4438, 4442, 4872, 4886
\l__enumext_columns_viii_int . 4392, 4395, 4399,
 4408, 4468, 4472, 4475, 4481, 4487, 4491, 5143, 5158
\l__enumext_counter_i_tl 41, 594
\l__enumext_counter_ii_tl 41, 595
\l__enumext_counter_iii_tl 41, 596
\l__enumext_counter_iv_tl 41, 597
\c__enumext_counter_style_tl 33, 46, 227
\g__enumext_counter_styles_tl . 28, 41, 63, 605,
 623
\l__enumext_counter_v_tl 41, 598, 859
\l__enumext_counter_vi_tl 41, 599
\l__enumext_counter_vii_tl 41, 600, 791
\l__enumext_counter_viii_tl 41, 601, 807
\l__enumext_current_widest_dim 28, 63, 629, 725,
 771, 840, 844
__enumext_def_meta_key:nnn .. 134, 5372, 5400,
 5406, 5420
__enumext_default_item:n ... 3240, 3240, 3304
__enumext_define_counters:Nn 28, 585, 585, 594,
 595, 596, 597, 598, 599, 600, 601
__enumext_endminipage: . 36, 375, 384, 579, 4570,
 4866, 5137
\g__enumext_envir_name_tl 34, 30, 284, 299, 358,
 2082, 2087, 2097, 2237, 2242, 2247, 2261, 2266, 2271
\l__enumext_envir_name_tl . 33, 34, 30, 253, 263,
 312, 320, 328, 5768, 5771, 5778, 5781, 5788, 5791,
 5798, 5801, 5807, 5811, 5817, 5821, 5878, 5882
__enumext_execute_after_env: 35, 69, 72, 83, 87,
 3020, 3020, 3838, 4899
__enumext_fake_item_indent: . 1051, 1051, 3603
\l__enumext_fake_item_indent_v_dim 1070, 1075
\l__enumext_fake_item_indent_v_tl 1072, 3444,
 3448, 3456
__enumext_fake_item_indent_vii: . 1051, 1080,
 3636
\l__enumext_fake_item_indent_vii_dim . 1083,
 1087
\l__enumext_fake_item_indent_vii_tl .. 1085,
 4861
__enumext_fake_item_indent_viii: 1051, 1092,
 3641
\l__enumext_fake_item_indent_viii_dim 1095,
 1099
\l__enumext_fake_item_indent_viii_tl . 1097,
 5128
\l__enumext_fake_item_indent_X_tl 96
__enumext_fake_make_label_vii:n . 122, 4796,
 4796, 4853
__enumext_fake_make_label_viii:n 5070, 5088,
 5120
__enumext_filter_first_level:n .. 132, 5269,
 5269, 5303, 5314
__enumext_filter_first_level_key:n 132, 5269,
 5274, 5278
__enumext_filter_first_level_pair:nn . 132,
 5269, 5275, 5287
__enumext_filter_save_key:n .. 76, 2365, 2373,
 2396, 2402, 2404, 2404, 5200, 5204, 5208, 5212, 5216,
 5220
__enumext_filter_save_key_key:n .. 76, 2404,
 2409, 2413
__enumext_filter_save_key_pair:nn 76, 2404,
 2410, 2421
__enumext_filter_series:n 64, 1839, 1839, 1877,
 1889, 1894
__enumext_filter_series_key:n 64, 1839, 1844,
 1848
__enumext_filter_series_pair:nn .. 64, 1839,
 1845, 1856
__enumext_first_item_tmp_vii: 119, 121, 4652,
 4725, 4725
__enumext_first_item_tmp_viii: .. 126, 4914,
 4968, 4968
\g__enumext_footnote_standar_arg_seq .. 169,
 453, 464, 467
\g__enumext_footnote_standar_int 169, 447, 450,
 452, 455
\g__enumext_footnote_standar_int_seq .. 169,
 455, 460, 463, 468
\g__enumext_footnote_starred_arg_seq .. 169,

```

483, 494, 497  
 \g\_\_enumext\_footnote\_starred\_int 169, 477, 480, 482, 485  
 \g\_\_enumext\_footnote\_starred\_int\_seq .. 169, 485, 490, 493, 498  
 \\_\_enumext\_footnotes\_key\_bool ..... 36  
 \l\_\_enumext\_footnotes\_key\_bool 31, 36, 159, 396, 401, 408, 509, 525, 539, 552  
 \\_\_enumext\_footnotetext:nn .. 436, 436, 465, 495  
 \\_\_enumext\_foreach\_add\_body:n . 135, 5421, 5481, 5484  
 \l\_\_enumext\_foreach\_after\_tl ..... 5425, 5493  
 \l\_\_enumext\_foreach\_before\_tl .... 5423, 5488  
 \g\_\_enumext\_foreach\_default\_keys\_tl 134, 122, 5443, 5464  
 \\_\_enumext\_foreach\_keyans:nn .. 135, 5421, 5460, 5462  
 \l\_\_enumext\_foreach\_name\_prop\_tl . 122, 5466, 5491  
 \l\_\_enumext\_foreach\_print\_seq 122, 5476, 5482, 5486  
 \l\_\_enumext\_foreach\_sep\_tl ..... 5435, 5482  
 \l\_\_enumext\_foreach\_start\_int .... 5427, 5478  
 \l\_\_enumext\_foreach\_step\_int ..... 5431, 5479  
 \l\_\_enumext\_foreach\_stop\_int . 5429, 5471, 5473, 5480  
 \\_\_enumext\_foreach\_wrapper:n ..... 5433, 5489  
 \\_\_enumext\_getkeyans:nn .. 130, 5169, 5183, 5187  
 \\_\_enumext\_getkeyans\_aux:n 130, 5169, 5171, 5174  
 \l\_\_enumext\_hyperref\_bool 31, 36, 159, 393, 411, 428, 2651, 3143, 4805  
 \\_\_enumext\_hypertarget:nn 36, 386, 413, 417, 433  
 \\_\_enumext\_if\_is\_int:n ..... 219  
 \\_\_enumext\_if\_is\_int:nTF ..... 219, 877, 891  
 \\_\_enumext\_internal\_mini\_page: 39, 100, 119, 566, 566, 3651, 4665  
 \\_\_enumext\_is\_not\_nested: . 28, 33, 100, 119, 247, 247, 3650, 4664  
 \\_\_enumext\_is\_on\_first\_level: . 28, 34, 100, 119, 247, 273, 3657, 4677  
 \g\_\_enumext\_item\_anskey\_int 82, 90, 149, 345, 372, 373, 2222, 2600, 3157  
 \\_\_enumext\_item\_answer\_diff: 72, 87, 2218, 2218, 3027  
 \g\_\_enumext\_item\_answer\_diff\_int 72, 149, 346, 2220, 2227, 2251  
 \l\_\_enumext\_item\_column\_pos\_vii\_int 121, 4426, 4432, 4438, 4442, 4449, 4736, 4872, 4875  
 \l\_\_enumext\_item\_column\_pos\_viii\_int .. 126, 4475, 4481, 4487, 4491, 4498, 4979, 5143, 5146  
 \l\_\_enumext\_item\_column\_pos\_X\_int ..... 175  
 \g\_\_enumext\_item\_count\_all\_vii\_int 121, 4450, 4737, 4886, 4894  
 \g\_\_enumext\_item\_count\_all\_viii\_int 126, 4499, 4980, 5157, 5166  
 \g\_\_enumext\_item\_count\_all\_X\_int ..... 175  
 \g\_\_enumext\_item\_number\_bool ..... 149  
 \l\_\_enumext\_item\_number\_bool 71, 157, 2172, 2177, 2181, 2185, 2198, 2775, 2829, 3247, 3281, 4814  
 \g\_\_enumext\_item\_number\_int . 71, 149, 344, 371, 373, 2171, 2176, 2180, 2184, 2197, 2222, 3246, 3280, 4813  
 \\_\_enumext\_item\_peek\_args\_vii: 121, 4733, 4738, 4740  
 \\_\_enumext\_item\_peek\_args\_viii: .. 126, 4976, 4981, 4983  
 \\_\_enumext\_item\_star\_exec: 93, 3259, 3286, 3331, 3352  
 \l\_\_enumext\_item\_star\_exec\_int 149, 3470, 3477, 3504, 4250, 4269, 4999, 5014, 5076  
 \l\_\_enumext\_item\_star\_wrap\_bool ..... 149  
 \l\_\_enumext\_item\_starred\_vii\_bool 4755, 4769, 4818  
 \l\_\_enumext\_item\_starred\_viii\_bool 4998, 5013, 5098, 5129  
 \l\_\_enumext\_item\_starred\_X\_bool ..... 175  
 \\_\_enumext\_item\_std:w . 36, 92, 96, 375, 379, 3250, 3256, 3284, 3444, 3448, 3456  
 \g\_\_enumext\_item\_symbol\_aux\_tl . 92, 126, 3264, 3267, 3292, 3336, 3356  
 \g\_\_enumext\_item\_symbol\_aux\_vii\_tl 4777, 4820, 4823, 4827, 4829  
 \g\_\_enumext\_item\_symbol\_aux\_X\_tl ..... 175  
 \l\_\_enumext\_item\_symbol\_sep\_vii\_dim .. 4785, 4792, 4826, 4828  
 \l\_\_enumext\_item\_symbol\_vii\_tl ..... 4823  
 \l\_\_enumext\_item\_text\_vii\_box .... 4845, 4870  
 \l\_\_enumext\_item\_text\_viii\_box ... 5112, 5141  
 \l\_\_enumext\_item\_text\_X\_box ..... 175  
 \l\_\_enumext\_item\_width\_vii\_dim ... 4365, 4374, 4453, 4461, 4462  
 \l\_\_enumext\_item\_width\_viii\_dim .. 4396, 4405, 4502, 4510, 4511  
 \l\_\_enumext\_item\_width\_X\_dim ..... 175  
 \l\_\_enumext\_itemindent\_X\_dim ..... 67  
 \l\_\_enumext\_itemsep\_i\_skip ... 1366, 1373, 1376, 1378, 1385, 1389, 1392, 1394, 1534, 1541, 1543, 1544, 1549, 1553, 1555, 1556  
 \l\_\_enumext\_itemsep\_ii\_skip .. 1406, 1413, 1416, 1418, 1425, 1429, 1432, 1434  
 \l\_\_enumext\_itemsep\_iii\_skip . 1445, 1452, 1455, 1457, 1464, 1468, 1471, 1473  
 \l\_\_enumext\_itemsep\_vii\_skip ..... 4892  
 \l\_\_enumext\_itemsep\_viii\_skip ..... 5164  
 \l\_\_enumext\_joined\_item\_aux\_vii\_int .. 4447, 4448, 4449, 4450, 4456  
 \l\_\_enumext\_joined\_item\_aux\_viii\_int . 4496, 4497, 4498, 4499, 4505  
 \l\_\_enumext\_joined\_item\_aux\_X\_int .... 175  
 \\_\_enumext\_joined\_item\_vii:w .. 121, 4733, 4743, 4744, 4746  
 \l\_\_enumext\_joined\_item\_vii\_int .. 4418, 4419, 4422, 4424, 4430, 4435, 4440, 4445, 4447, 4453  
 \\_\_enumext\_joined\_item\_viii:w . 126, 4976, 4986, 4987, 4989  
 \l\_\_enumext\_joined\_item\_viii\_int . 4467, 4468, 4471, 4473, 4479, 4484, 4489, 4494, 4496, 4502  
 \l\_\_enumext\_joined\_item\_X\_int ..... 175  
 \l\_\_enumext\_joined\_width\_vii\_dim . 4451, 4458, 4461, 4847, 4855  
 \l\_\_enumext\_joined\_width\_viii\_dim 4500, 4507, 4510, 5114, 5122  
 \l\_\_enumext\_joined\_width\_X\_dim ..... 175  
 \\_\_enumext\_keyans\_addto\_prop:n 88, 3040, 3040, 3459, 4252  
 \\_\_enumext\_keyans\_addto\_seq:n . 89, 3116, 3116, 3461, 4254  
 \\_\_enumext\_keyans\_addto\_seq\_link: 3116, 3137,



3139, 5048  
 \\_\_enumext\_keyans\_default\_item:n .. 96, 3439, 3439, 3478  
 \l\_\_enumext\_keyans\_env\_bool 30, 3682, 3695, 3847, 3937  
 \\_\_enumext\_keyans\_fake\_item\_indent: .. 1051, 1067, 3593  
 \l\_\_enumext\_keyans\_level\_h\_int .. 125, 24, 800, 826, 2759, 2821, 3094, 4671, 4931, 4932  
 \l\_\_enumext\_keyans\_level\_int .. 24, 1654, 2755, 2817, 3089, 3846, 3851, 4216  
 \\_\_enumext\_keyans\_make\_label: . 97, 3482, 3482, 3591  
 \\_\_enumext\_keyans\_make\_label\_box: 3482, 3486, 3491, 3526  
 \\_\_enumext\_keyans\_make\_label\_std: 3482, 3494, 3516  
 \\_\_enumext\_keyans\_mini\_right\_cmd:n 61, 1656, 1689, 1689  
 \\_\_enumext\_keyans\_mini\_set\_vskip: ..... 57  
 \\_\_enumext\_keyans\_minipage\_add\_space: 1485, 1511, 3876  
 \\_\_enumext\_keyans\_minipage\_set\_skip: . 1485, 1485, 1513  
 \\_\_enumext\_keyans\_multi\_addvspace: 1285, 1296, 3900  
 \\_\_enumext\_keyans\_multi\_set\_vskip: 54, 1285, 1285, 1298  
 \\_\_enumext\_keyans\_multicols\_start: 3864, 3879, 3881  
 \\_\_enumext\_keyans\_multicols\_stop: 1693, 3864, 3906, 3935  
 \\_\_enumext\_keyans\_name\_and\_start: 28, 34, 125, 306, 306, 3848, 4105, 4936  
 \\_\_enumext\_keyans\_parse\_keys:n 3860, 3860, 3955  
 \\_\_enumext\_keyans\_pic\_arg\_two: 109, 4098, 4121, 4151  
 \l\_\_enumext\_keyans\_pic\_level\_int .. 24, 1635, 2763, 2825, 3043, 3084, 3119, 3207, 4100, 4101  
 \\_\_enumext\_keyans\_pic\_parse\_keys:n 4098, 4107, 4150  
 \g\_\_enumext\_keyans\_pic\_parsep\_skip ... 138  
 \\_\_enumext\_keyans\_pic\_safe\_exec: . 109, 4098, 4098, 4149  
 \\_\_enumext\_keyans\_pic\_skip\_abs:N . 109, 4098, 4114, 4125  
 \\_\_enumext\_keyans\_pre\_itemsep\_skip: .. 1485, 1504, 1531  
 \\_\_enumext\_keyans\_redefine\_item: .. 96, 3464, 3464, 3590  
 \\_\_enumext\_keyans\_ref: ..... 45, 851, 868, 3592  
 \\_\_enumext\_keyans\_ref:n ..... 45, 848, 851, 851  
 \\_\_enumext\_keyans\_safe\_exec: . 3840, 3840, 3954  
 \\_\_enumext\_keyans\_set\_item\_width: 106, 3941, 3941, 3963  
 \\_\_enumext\_keyans\_show\_ans: .. 3160, 3168, 3187  
 \\_\_enumext\_keyans\_show\_item\_opt: .. 96, 3160, 3175, 3457, 4266, 5131  
 \\_\_enumext\_keyans\_show\_left:n . 96, 3160, 3160, 3454, 4261  
 \\_\_enumext\_keyans\_show\_pos: .. 3160, 3172, 3200  
 \\_\_enumext\_keyans\_starred\_item:n .. 96, 3451, 3451, 3473  
 \\_\_enumext\_keyans\_store\_ref: .. 88, 3063, 3063, 3460, 4253, 5046  
 \\_\_enumext\_keyans\_store\_ref\_aux\_i: 89, 3063, 3075, 3078  
 \\_\_enumext\_keyans\_store\_ref\_aux\_ii: 89, 3063, 3104, 3106  
 \\_\_enumext\_keyans\_unknown\_keys:n . 3377, 3381, 3385, 4096  
 \\_\_enumext\_keyans\_unknown\_keys:nn 3377, 3387, 3389  
 \\_\_enumext\_keyans\_wraper\_label:n .. 97, 3482, 3498, 3522, 3535, 4265  
 \\_\_enumext\_keyans\_wraper\_label\_viii:n 5070, 5070, 5105  
 \\_\_enumext\_keyans\_wrapper\_item:n . 2303, 2342, 3505, 3508, 5077, 5080  
 \\_\_enumext\_keyans\_wrapper\_opt:n .. 2300, 2340, 3183  
 \l\_\_enumext\_label\_copy\_i\_tl .. 2566, 3082, 3087, 3092, 3097  
 \l\_\_enumext\_label\_copy\_v\_tl ..... 3092  
 \l\_\_enumext\_label\_copy\_vi\_tl ..... 3087  
 \l\_\_enumext\_label\_copy\_vii\_tl 2542, 2553, 2582, 3082  
 \l\_\_enumext\_label\_copy\_viii\_tl ..... 3097  
 \l\_\_enumext\_label\_copy\_X\_tl ..... 161  
 \l\_\_enumext\_label\_fill\_left\_v\_tl ..... 3520  
 \l\_\_enumext\_label\_fill\_left\_X\_tl ..... 96  
 \l\_\_enumext\_label\_fill\_right\_v\_tl .... 3523  
 \l\_\_enumext\_label\_fill\_right\_X\_tl ..... 96  
 \l\_\_enumext\_label\_font\_style\_v\_tl 3521, 3534, 4264, 4270  
 \l\_\_enumext\_label\_font\_style\_vii\_tl ... 4833  
 \l\_\_enumext\_label\_font\_style\_viii\_tl .. 5104  
 \l\_\_enumext\_label\_i\_tl ..... 717  
 \l\_\_enumext\_label\_ii\_tl ..... 717  
 \l\_\_enumext\_label\_iii\_tl ..... 717  
 \l\_\_enumext\_label\_iv\_tl ..... 717  
 \\_\_enumext\_label\_style:Nnn 28, 41, 618, 618, 633, 722, 768, 837, 841  
 \l\_\_enumext\_label\_v\_tl 89, 834, 3048, 3124, 3194, 3234, 3453, 3458, 3958, 4129, 4260, 4262  
 \l\_\_enumext\_label\_vi\_tl 89, 834, 3045, 3121, 4260, 4262, 4265, 4271  
 \l\_\_enumext\_label\_vii\_tl . 763, 4764, 4787, 4794  
 \l\_\_enumext\_label\_viii\_tl 763, 5008, 5040, 5044  
 \l\_\_enumext\_label\_width\_by\_box .. 63, 614, 615  
 \\_\_enumext\_label\_width\_by\_box:Nn 41, 612, 612, 617, 629, 901  
 \l\_\_enumext\_labelsep\_i\_dim ... 3192, 3197, 3205, 3237, 5052, 5067  
 \l\_\_enumext\_labelsep\_v\_dim ..... 3890  
 \l\_\_enumext\_labelsep\_vii\_dim . 2667, 3192, 3205, 4360, 4370, 4454, 4729, 4785, 4840, 4849  
 \l\_\_enumext\_labelsep\_viii\_dim 4391, 4401, 4503, 4972, 5107, 5116  
 \l\_\_enumext\_labelwidth\_i\_dim . 3191, 3197, 3204, 3237, 5052, 5067  
 \l\_\_enumext\_labelwidth\_v\_dim ..... 3532, 3890  
 \l\_\_enumext\_labelwidth\_vii\_dim ... 2667, 3191, 3204, 4360, 4369, 4454, 4729, 4831, 4848  
 \l\_\_enumext\_labelwidth\_viii\_dim .. 4391, 4400, 4503, 4972, 5102, 5115  
 \l\_\_enumext\_leftmargin\_tmp\_v\_bool . 109, 4123  
 \l\_\_enumext\_leftmargin\_tmp\_X\_bool ..... 67



`\l__enumext_leftmargin_tmp_X_dim` . . . . . 67  
`\l__enumext_leftmargin_X_dim` . . . . . 67  
`\__enumext_level:` 215, 215, 746, 749, 750, 758, 760, 1054, 1058, 1062, 1130, 1134, 1138, 1142, 1225, 1227, 1229, 1231, 1273, 1275, 1277, 1279, 1283, 1317, 1323, 1328, 1330, 1333, 1336, 1349, 1352, 1663, 1667, 1673, 1736, 1738, 1740, 1743, 1750, 1752, 1754, 1757, 2360, 2362, 2364, 2392, 2393, 2395, 2451, 2459, 2463, 2467, 2671, 2672, 3249, 3250, 3254, 3255, 3256, 3264, 3272, 3273, 3276, 3283, 3284, 3288, 3291, 3293, 3327, 3328, 3329, 3332, 3335, 3346, 3347, 3349, 3350, 3353, 3688, 3701, 3708, 3716, 3719, 3721, 3723, 3724, 3725, 3726, 3729, 3734, 3740, 3746, 3753, 3766, 3768, 3771, 3772, 3774, 3778, 3784, 3809, 3814, 3825, 3827  
`\l__enumext_level_h_int` 119, 24, 256, 279, 293, 784, 819, 1642, 2168, 2188, 2561, 2795, 2807, 3696, 4666, 4667  
`\l__enumext_level_int` . 100, 24, 217, 266, 278, 294, 568, 1237, 1362, 1641, 2162, 2194, 2538, 2548, 2554, 2560, 2567, 2576, 2581, 2794, 2806, 3022, 3607, 3652, 3653, 3664, 3672, 3686, 3699, 3730, 3855, 4212, 4709, 4719, 4944, 5808, 5812, 5818, 5822  
`\__enumext_list_arg_two_i:` . . . . . 3572  
`\__enumext_list_arg_two_ii:` . . . . . 3572  
`\__enumext_list_arg_two_iii:` . . . . . 3572  
`\__enumext_list_arg_two_iv:` . . . . . 3572  
`\__enumext_list_arg_two_v:` . 96, 3572, 3960, 4124  
`\__enumext_list_arg_two_vii:` . . . . 3613, 4646  
`\__enumext_list_arg_two_viii:` . . . . 3613, 4908  
`\l__enumext_listoffset_v_dim` . 3892, 3946, 3949  
`\l__enumext_listparindent_vii_dim` 4856, 4860  
`\l__enumext_listparindent_viii_dim` 5123, 5127  
`\__enumext_log_answer_vars:` . 35, 360, 368, 3029  
`\__enumext_log_global_vars:` . 35, 360, 360, 3028  
`\__enumext_make_label:` . . . . 93, 3307, 3307, 3601  
`\__enumext_make_label_box:` . . . 3307, 3311, 3316, 3339  
`\__enumext_make_label_std:` . . . 3307, 3319, 3323  
`\l__enumext_mark_answer_sym_tl` 78, 2308, 2517, 2684, 3209, 3222, 5056  
`\l__enumext_mark_position_str` 126, 2312, 2313, 2345, 2346, 2515  
`\l__enumext_mark_ref_sym_tl` . . 2325, 2656, 3151  
`\l__enumext_meta_path_tl` . 122, 5396, 5397, 5399, 5400  
`\c__enumext_meta_paths_prop` . . . . . 134, 5372  
`\__enumext_mini_addvspace_vii:` 60, 1621, 1621, 4528  
`\__enumext_mini_addvspace_viii:` 60, 1621, 1627, 4593  
`\__enumext_mini_env*` . . . . . 566  
`\__enumext_mini_page` 1673, 1700, 3778, 3877, 4530, 4595, 4616  
`\__enumext_mini_right_cmd:n` . 60, 61, 1658, 1660, 1660  
`\__enumext_mini_set_vskip_vii:` 59, 1564, 1564, 1623  
`\__enumext_mini_set_vskip_viii:` 59, 1564, 1586, 1629  
`\__enumext_minipage:w` 36, 375, 383, 573, 4553, 4855, 5122  
`\l__enumext_minipage_active_v_bool` 3874, 3897, 3922  
`\g__enumext_minipage_active_vii_bool` . . 117, 4542, 4551, 4573  
`\l__enumext_minipage_active_vii_bool` . 4524, 4535  
`\g__enumext_minipage_active_viii_bool` 4606, 4614, 4633  
`\l__enumext_minipage_active_viii_bool` 4589, 4600  
`\g__enumext_minipage_active_X_bool` . . . 175  
`\l__enumext_minipage_active_X_bool` . . . . 83  
`\__enumext_minipage_add_space:` . 55, 103, 1313, 1339, 3776  
`\g__enumext_minipage_after_skip` 83, 1568, 1580, 4571, 4631  
`\l__enumext_minipage_after_skip` . . 55, 103, 83, 1326, 1366, 1368, 1373, 1376, 1380, 1385, 1389, 1392, 1396, 1408, 1413, 1416, 1420, 1425, 1429, 1432, 1436, 1447, 1452, 1455, 1459, 1464, 1468, 1471, 1475, 1487, 1501, 1534, 1536, 1541, 1543, 1545, 1549, 1553, 1555, 1557, 1588, 1601, 1615, 1669, 1696, 3932  
`\g__enumext_minipage_center_vii_bool` . 4557, 4574  
`\g__enumext_minipage_center_viii_bool` 4618, 4634  
`\g__enumext_minipage_center_X_bool` . . . 175  
`\l__enumext_minipage_hsep_v_dim` . . . . . 3872  
`\l__enumext_minipage_hsep_vii_dim` . . . . 4522  
`\l__enumext_minipage_hsep_viii_dim` . . . 4587  
`\l__enumext_minipage_left_skip` 83, 1488, 1566, 1571, 1575, 1589, 1593, 1607, 1625, 1631  
`\l__enumext_minipage_left_v_dim` . . 3870, 3877  
`\l__enumext_minipage_left_vii_dim` 4518, 4530  
`\l__enumext_minipage_left_viii_dim` 4583, 4595  
`\l__enumext_minipage_left_X_dim` . . . . . 83  
`\g__enumext_minipage_right_skip` 83, 1567, 1572, 1576, 4556, 4617  
`\l__enumext_minipage_right_skip` . 55, 83, 1315, 1321, 1326, 1328, 1330, 1489, 1490, 1496, 1501, 1502, 1503, 1508, 1590, 1597, 1611, 1675, 1702  
`\l__enumext_minipage_right_v_dim` . 1691, 1700, 3868, 3872  
`\g__enumext_minipage_right_vii_dim` 117, 4526, 4553, 4576  
`\l__enumext_minipage_right_vii_dim` 117, 4516, 4521, 4527  
`\g__enumext_minipage_right_viii_dim` . . 4591, 4616, 4636  
`\l__enumext_minipage_right_viii_dim` . . 4581, 4586, 4592  
`\g__enumext_minipage_right_X_dim` . . . . . 175  
`\g__enumext_minipage_right_X_skip` . . . . 175  
`\__enumext_minipage_set_skip:` . 55, 1313, 1313, 1341  
`\g__enumext_minipage_stat_int` . . 103, 83, 1680, 1707, 3775, 3786, 3791, 3875, 3924, 3929  
`\l__enumext_minipage_temp_skip` 83, 1387, 1397, 1400, 1427, 1437, 1440, 1466, 1476, 1479, 1551, 1558, 1560  
`\l__enumext_miniright_code_vii_box` 4564, 4568  
`\g__enumext_miniright_code_vii_tl` 117, 4559, 4566, 4575  
`\l__enumext_miniright_code_viii_box` . . 4625, 4629  
`\g__enumext_miniright_code_viii_tl` 4620, 4627, 4635  
`\l__enumext_miniright_code_X_box` . . . . . 175

`\l__enumext_mode_box_bool` . . . . 638, 3314, 3489  
`\__enumext_multi_addvspace:` 53, 102, 1268, 1268, 3737  
`\__enumext_multi_set_vskip:` 53, 1223, 1223, 1270  
`\l__enumext_multicols_above_ii_skip` . . . 1242  
`\l__enumext_multicols_above_iii_skip` . . 1251  
`\l__enumext_multicols_above_iv_skip` . . . 1260  
`\l__enumext_multicols_above_v_skip` 1287, 1301, 1311, 1502  
`\l__enumext_multicols_above_X_skip` . . . . 75  
`\l__enumext_multicols_below_ii_skip` . . 1369, 1378, 1382, 1394, 1399  
`\l__enumext_multicols_below_iii_skip` . 1409, 1418, 1422, 1434, 1439  
`\l__enumext_multicols_below_iv_skip` . . 1448, 1457, 1461, 1473, 1478  
`\l__enumext_multicols_below_v_skip` 1291, 1305, 1503, 1537, 1544, 1546, 1556, 1559, 3914  
`\l__enumext_multicols_below_X_skip` . . . . 75  
`\g__enumext_multicols_right_X_skip` . . . . 75  
`\__enumext_multicols_start:` 102, 103, 3713, 3713, 3780  
`\__enumext_multicols_stop:` 102, 1665, 3743, 3743, 3796  
`\__enumext_nested_base_line_fix:` 48, 100, 975, 981, 3668  
`\__enumext_newlabel:nn` 31, 37, 79, 421, 421, 2592, 3110  
`\l__enumext_newlabel_arg_one_tl` 31, 37, 79, 89, 161, 2585, 2593, 2655, 3099, 3111, 3149  
`\l__enumext_newlabel_arg_two_tl` 31, 37, 78, 161, 2541, 2551, 2564, 2579, 2594, 3086, 3091, 3096, 3112  
`\__enumext_parse_foreach_keys:n` . . 5421, 5437, 5454  
`\__enumext_parse_foreach_keys:nn` . 5421, 5444, 5456  
`\__enumext_parse_keys:n` 48, 64, 3659, 3659, 3821  
`\__enumext_parse_keys_vii:n` 64, 4641, 4679, 4679  
`\__enumext_parse_keys_viii:n` . 4904, 4949, 4949  
`\__enumext_parse_save_key:n` 75, 2385, 2390, 2390  
`\__enumext_parse_save_key_vii:n` 75, 2380, 2390, 2398  
`\__enumext_parse_series:n` . . 64, 100, 120, 1865, 1865, 3667, 4685  
`\__enumext_parse_store_keys:n` . . . . . 100  
`\l__enumext_parsep_i_skip` . . . . . 1240, 1244  
`\l__enumext_parsep_ii_skip` . . . . . 1249, 1253  
`\l__enumext_parsep_iii_skip` . . . . . 1258, 1262  
`\l__enumext_parsep_vii_skip` . . . . . 4857  
`\l__enumext_parsep_viii_skip` . . . . . 5124  
`\l__enumext_partopsep_v_skip` . 1303, 1307, 1498, 1521  
`\l__enumext_partopsep_viii_skip` . . . . . 1599  
`\__enumext_phantomsection:` 36, 386, 414, 418, 434  
`\__enumext_pre_itemsep_skip:` 55, 56, 1331, 1360, 1360  
`\__enumext_print_footnote:` . . 436, 458, 522, 527  
`\__enumext_print_footnote_mini:` 436, 488, 549, 554  
`\__enumext_print_footnote_standar:` 500, 516, 580  
`\__enumext_print_footnote_starred:` 500, 545, 560, 564  
`\__enumext_print_keyans_box:NN` 78, 2509, 2509, 2522, 2666, 2670, 3196, 3236, 5052, 5067  
`\l__enumext_print_keyans_i_tl` . . . . 5205, 5227  
`\l__enumext_print_keyans_ii_tl` . . . 5209, 5228  
`\l__enumext_print_keyans_iii_tl` . . 5213, 5229  
`\l__enumext_print_keyans_iv_tl` . . . 5217, 5230  
`\l__enumext_print_keyans_star_bool` . 48, 131, 126, 987, 995, 5250, 5255  
`\l__enumext_print_keyans_starred_tl` 130, 131, 126, 5201, 5248  
`\l__enumext_print_keyans_vii_tl` 130, 5221, 5231  
`\l__enumext_print_keyans_X_tl` . . . . . 126  
`\__enumext_printkeyans:nnn` 131, 5224, 5232, 5235  
`\__enumext_redefine_item:` . 93, 3296, 3296, 3600  
`\l__enumext_ref_key_arg_tl` 43, 46, 230, 739, 740, 752, 783, 786, 796, 802, 812, 853, 854, 864  
`\l__enumext_ref_the_count_tl` . 43, 46, 746, 749, 752, 791, 793, 796, 807, 809, 812, 859, 861, 864  
`\__enumext_regex_counter_style:` . . 33, 43, 225, 225, 747, 792, 808, 860  
`\__enumext_register_counter_style:Nn` . . 602, 602, 607, 608, 609, 610, 611  
`\__enumext_remove_extra_parsep_vii:` . . 4659, 4881, 4881  
`\__enumext_remove_extra_parsep_viii:` . 4921, 5152, 5152  
`\__enumext_renew_footnote:` . . 436, 440, 506, 511  
`\__enumext_renew_footnote_mini:` 436, 470, 536, 541  
`\__enumext_renew_footnote_standar:` 500, 500, 572  
`\__enumext_renew_footnote_starred:` 500, 532, 4851, 5118  
`\l__enumext_renew_the_count_v_tl` 862, 870, 872  
`\l__enumext_renew_the_count_vii_tl` 794, 821, 823  
`\l__enumext_renew_the_count_viii_tl` 810, 828, 830  
`\l__enumext_renew_the_count_X_tl` . . . . . 46  
`\__enumext_rescan_anskey_env:n` . . 85, 87, 2838, 2904, 2999, 3007  
`\__enumext_reset_global_bool:` . . 336, 339, 348  
`\__enumext_reset_global_int:` . . . 336, 338, 342  
`\__enumext_reset_global_tl:` . . . . 336, 340, 354  
`\__enumext_reset_global_vars:` . 35, 87, 336, 336, 3037  
`\l__enumext_resume_active_bool` 64, 67, 57, 1869, 1989  
`\__enumext_resume_counter:` . . 66, 67, 1987, 1993, 2000  
`\__enumext_resume_counter:n` . 64, 67, 1958, 1963, 1987, 1987, 2057, 2065  
`\__enumext_resume_counter_save_ans:` . . 67, 68, 1987, 1998, 2030  
`\__enumext_resume_counter_series:` . 67, 1987, 1996, 2013  
`\g__enumext_resume_int` . . . 57, 1910, 2004, 2005  
`\__enumext_resume_last:n` 64, 65, 1865, 1871, 1884  
`\l__enumext_resume_name_tl` 57, 1906, 1914, 1917, 1933, 1941, 1944, 1990, 1991, 2019, 2026  
`\__enumext_resume_save_counter:` . 65, 103, 120, 1897, 1897, 3802, 4703  
`\__enumext_resume_series:n` . 66, 1833, 1954, 1954  
`\__enumext_resume_starred:` . 68, 1834, 2051, 2051  
`\g__enumext_resume_vii_int` 57, 1937, 2009, 2010

```

\l__enumext_rightmargin_vii_dim .. 4372, 4376,
 4381
\l__enumext_rightmargin_viii_dim . 4403, 4407,
 4412
__enumext_safe_exec: .. 39, 100, 3648, 3648, 3820
__enumext_safe_exec_vii: . 39, 4640, 4662, 4662
__enumext_safe_exec_viii: 125, 4903, 4925, 4925
__enumext_second_part: .. 103, 3782, 3782, 3834
__enumext_second_part_v: ... 3864, 3920, 3968
\l__enumext_series_name_tl 67
\l__enumext_series_str . 65, 100, 120, 1831, 1867,
 1875, 1876, 1878, 1880, 1901, 1904, 1908, 1928, 1931,
 1935, 3663, 4683
__enumext_set_error:nn 5331, 5368, 5370
__enumext_set_item_width: 103, 3804, 3804, 3830
__enumext_set_parse:n 5331, 5342, 5358
\l__enumext_setkey_tmpa_int ... 117, 5335, 5339
\l__enumext_setkey_tmpa_seq .. 117, 5333, 5343,
 5349, 5351, 5353, 5365
\l__enumext_setkey_tmpa_tl ... 117, 5341, 5345
\l__enumext_setkey_tmpb_seq .. 117, 5334, 5337,
 5341, 5342
\l__enumext_setkey_tmpb_tl 117, 5360, 5362, 5363
\l__enumext_show_answer_bool . 2319, 2349, 2678,
 3166, 3180, 3503, 4257, 5050, 5075
__enumext_show_length:nnn .. 50, 233, 233, 5579,
 5580, 5581, 5582, 5583, 5584, 5585, 5586, 5587, 5588,
 5594, 5595, 5596, 5597, 5598, 5599, 5600, 5601, 5602,
 5603
\l__enumext_show_position_bool ... 2322, 2352,
 2682, 3170, 3181, 4258, 5054
\g__enumext_standar_bool 33, 100, 30, 255, 258, 277,
 351, 502, 518, 1899, 1964, 1976, 2002, 2015, 2053,
 2193, 2207, 2546, 2559, 2574, 3683
\l__enumext_standar_bool 100, 103, 30, 1649, 2547,
 3655, 3801, 4676
\l__enumext_standar_first_bool 34, 100, 30, 282,
 1886, 2033, 2095, 2102
__enumext_standar_item_vii:w . 121, 4733, 4751,
 4753
__enumext_standar_item_viii:w 126, 127, 4976,
 4994, 4996
__enumext_standar_ref: 43, 737, 756, 3602
__enumext_standar_ref:n 43, 729, 737, 737
\g__enumext_standar_series_tl . 57, 1888, 1889,
 2055, 2058
__enumext_standar_unknown_keys:n 3417, 3421,
 3425
__enumext_standar_unknown_keys:nn 3417, 3427,
 3429
\g__enumext_starred_bool 33, 119, 30, 265, 268, 292,
 352, 1648, 1926, 1969, 1980, 2007, 2022, 2061, 2167,
 2213, 2537, 3080, 4577
\l__enumext_starred_bool 119, 120, 125, 30, 2575,
 2610, 2616, 2664, 2953, 2958, 3189, 3202, 3656, 4675,
 4702, 4937, 4941
__enumext_starred_columns_set_vii: .. 4354,
 4354, 4650
__enumext_starred_columns_set_viii: . 4354,
 4385, 4912
\l__enumext_starred_first_bool 34, 119, 30, 297,
 985, 994, 1891, 2042, 2095, 2102
__enumext_starred_item:nn ... 3259, 3259, 3302
__enumext_starred_item_exec: . 127, 5011, 5042,
 5100
__enumext_starred_item_vii:w . 121, 122, 4733,
 4750, 4767
__enumext_starred_item_vii_aux_i:w .. 4733,
 4772, 4775
__enumext_starred_item_vii_aux_ii:w . 4733,
 4773, 4778, 4780
__enumext_starred_item_vii_aux_iii:w 4733,
 4783, 4790
__enumext_starred_item_viii:w 126, 127, 4993,
 5011, 5011
__enumext_starred_item_viii_aux_i:w .. 127,
 5011, 5017, 5020
__enumext_starred_item_viii_aux_ii:w . 127,
 5011, 5018, 5035, 5037
__enumext_starred_joined_item_vii:n 115, 121,
 4416, 4416, 4748
__enumext_starred_joined_item_viii:n . 115,
 126, 4416, 4465, 4991
__enumext_starred_ref: 44, 781, 817, 3633
__enumext_starred_ref:n 44, 775, 781, 781
\g__enumext_starred_series_tl . 57, 1893, 1894,
 2063, 2066
__enumext_starred_unknown_keys:n 3399, 3401,
 3403
__enumext_starred_unknown_keys:nn 3399, 3405,
 3407
__enumext_start_from:NNn 45, 875, 875, 888, 910,
 916
\l__enumext_start_i_int 2005, 2017, 2036
__enumext_start_item_tmp_vii: 119, 4653, 4733,
 4733
__enumext_start_item_tmp_viii: .. 4915, 4976,
 4976
__enumext_start_item_vii:w 121, 123, 4759, 4764,
 4787, 4794, 4842, 4842
__enumext_start_item_viii:w .. 127, 5003, 5008,
 5040, 5109, 5109
\g__enumext_start_line_tl 34, 30, 285, 300, 357,
 2237, 2242, 2247, 2261, 2266, 2271
__enumext_start_list:nn . 36, 98, 375, 377, 3824,
 3957, 4644, 4906
__enumext_start_list_tag:n .. 3970, 3996, 4852,
 5119
__enumext_start_mini_vii: 120, 4514, 4514, 4694
__enumext_start_mini_viii: ... 125, 4579, 4579,
 4960
__enumext_start_save_ans_msg: 69, 2079, 2079,
 2104
__enumext_start_store_level: . 101, 3677, 3677,
 3823
__enumext_start_store_level_vii: 120, 4643,
 4705, 4705
\l__enumext_start_vii_int ... 2010, 2024, 2045
\l__enumext_start_X_int 96
__enumext_stop_item_tmp_vii: .. 119, 121, 123,
 4652, 4658, 4735, 4844
__enumext_stop_item_tmp_viii: 126, 4914, 4920,
 4978, 5111
__enumext_stop_item_vii: 123, 124, 4842, 4844,
 4864
__enumext_stop_item_viii: ... 5109, 5111, 5135
__enumext_stop_list: 36, 117, 120, 375, 378, 3748,
 3756, 3910, 3917, 4537, 4545, 4602, 4609

```

```

__enumext_stop_list_tag:n ... 3970, 4012, 4867,
 5138
__enumext_stop_mini_vii: 117, 120, 4514, 4533,
 4698
__enumext_stop_mini_viii: 126, 4579, 4598, 4964
__enumext_stop_save_ans_msg: . 69, 2079, 2084,
 3026
__enumext_stop_start_list_tag: .. 3970, 4004,
 4854, 5121
__enumext_stop_store_level: .. 101, 102, 3706,
 3706, 3749, 3757
__enumext_stop_store_level_vii: .. 117, 120,
 4538, 4546, 4705, 4715
\l__enumext_store_active_bool 30, 69, 108, 2034,
 2043, 2111, 2751, 3681, 3694, 3842, 3850, 4208, 4707,
 4717, 4927, 4943
__enumext_store_active_keys:n 74, 75, 100, 2358,
 2358, 3674
__enumext_store_active_keys_vii:n 74, 75, 120,
 2358, 2368, 4686
__enumext_store_addto_prop:n 76, 88, 2433, 2433,
 2441, 2601, 3061, 5045
__enumext_store_addto_seq:n 76, 90, 2442, 2442,
 2446, 2453, 2467, 2475, 2484, 2498, 2506, 2659, 3154
\l__enumext_store_anskey_arg_tl .. 30, 80, 108,
 2607, 2612, 2614, 2619, 2626, 2629, 2639, 2644, 2647,
 2653, 2659
__enumext_store_anskey_code:n 79, 82, 87, 2598,
 2598, 2744, 2997, 3005
\l__enumext_store_anskey_env_tl .. 30, 86, 108,
 2927, 2931, 2937, 2999, 3007
\l__enumext_store_anskey_opt_tl .. 30, 86, 108,
 2928, 2955, 2961, 2968, 2974, 2984, 2994, 3003
__enumext_store_anskey_safe_outer: 82
\g__enumext_store_columns_break_bool . 2851,
 2952, 3014
\l__enumext_store_columns_break_bool . 2609,
 2700
\l__enumext_store_current_label_tl 30, 88-90,
 127, 108, 3042, 3045, 3048, 3054, 3059, 3061, 3118,
 3121, 3124, 3130, 3135, 3145, 3154, 5022, 5027, 5031,
 5044, 5045, 5047
\l__enumext_store_current_label_tmp_tl . 30,
 108, 3453, 3458
\l__enumext_store_current_opt_arg_tl 30, 127,
 108, 3164, 3177, 3183, 5033
__enumext_store_internal_ref: .. 78, 79, 2523,
 2523, 2604
\g__enumext_store_item_join_int .. 2854, 2959,
 2963, 3015
\l__enumext_store_item_join_int .. 2617, 2621,
 2703
\g__enumext_store_item_star_bool . 2856, 2966,
 3016
\l__enumext_store_item_star_bool . 2624, 2705
\g__enumext_store_item_symbol_sep_dim 2861,
 2981, 2986, 3018
\l__enumext_store_item_symbol_sep_dim 2636,
 2641, 2710
\g__enumext_store_item_symbol_tl . 2859, 2972,
 2976, 3017
\l__enumext_store_item_symbol_tl . 2627, 2631,
 2708
\l__enumext_store_keyans_item_opt_sep_-
 tl 2305, 2338, 3052, 3056, 3128, 3132, 5025, 5029
__enumext_store_level_close: . 77, 2447, 2471,
 3710
__enumext_store_level_close_vii: . 77, 2478,
 2502, 4721
__enumext_store_level_open: 77, 101, 2447, 2447,
 3689, 3702
__enumext_store_level_open_vii: .. 77, 2478,
 2478, 4711
\g__enumext_store_name_tl 30, 69, 108, 356, 363,
 364, 365, 366, 2087, 2113, 2236, 2241, 2246, 2260,
 2265, 2270, 3024
\l__enumext_store_name_tl 30, 69, 71, 108, 1920,
 1923, 1947, 1950, 2038, 2047, 2082, 2091, 2092, 2113,
 2114, 2115, 2117, 2118, 2120, 2122, 2123, 2125, 2127,
 2128, 2152, 2435, 2437, 2444, 2587, 2588, 2690, 2933,
 3101, 3102, 3215, 3228, 5062
\l__enumext_store_ref_key_bool 79, 2328, 2602,
 2650, 3065, 3142
\l__enumext_store_save_key_vii_bool .. 2370,
 2400
\l__enumext_store_save_key_vii_tl 2372, 2373,
 2401, 2402, 2482, 2490, 2494, 2498
\l__enumext_store_save_key_X_bool .. 74, 126
\l__enumext_store_save_key_X_tl 75, 126
\l__enumext_store_upper_level_X_bool .. 126
__enumext_storing_exec: 69, 84, 2089, 2105, 2109
__enumext_storing_set:n .. 69, 2074, 2089, 2089
\l__enumext_the_counter_v_tl 861
\l__enumext_the_counter_vii_tl 793
\l__enumext_the_counter_viii_tl 809
\l__enumext_the_counter_X_tl 46
__enumext_tmp:n 41, 45, 50, 56, 67, 74, 75, 82, 90, 95,
 96, 107, 130, 137, 164, 168, 175, 195, 634, 643, 1827,
 1838, 2070, 2078, 2131, 2149, 2290, 2333, 2334, 2357,
 2376, 2389, 2525, 2532, 2533, 2554, 2567, 2570, 2581,
 3067, 3074, 3377, 3384, 3417, 3424, 3572, 3612, 3613,
 3647
__enumext_tmp:nn 644, 665, 666, 700, 701, 716, 905,
 930, 1007, 1029, 1030, 1050, 1104, 1112, 1113, 1127,
 1192, 1208, 1209, 1222, 1716, 1732, 3361, 3376
__enumext_tmp:nnn 717, 733, 734, 735, 736, 763, 779,
 780
__enumext_tmp:nnnnn 931, 956, 959, 962, 964, 966,
 969, 972
__enumext_tmp:w 5180, 5183
\l__enumext_tmpa_vii_int 4364, 4367, 4376, 4407
\l__enumext_tmpa_viii_int 4395, 4398
\l__enumext_tmpa_X_dim 175
\l__enumext_tmpa_X_int 175
\l__enumext_topsep_v_skip 1289, 1293, 1492, 4201
\l__enumext_topsep_vii_skip .. 1569, 1578, 1582
\l__enumext_topsep_viii_skip . 1591, 1613, 1617
__enumext_undefine_anskey_env: . 83, 87, 2784,
 2784, 3035
__enumext_unskip_unkern: .. 33, 239, 239, 1342,
 1514, 3751, 3752, 3792, 3912, 3913, 3930, 4858, 4859,
 5125, 5126
\l__enumext_vspace_a_star_v_bool 1765
\l__enumext_vspace_a_star_vii_bool ... 1787
\l__enumext_vspace_a_star_viii_bool ... 1798
\l__enumext_vspace_a_star_X_bool 96
__enumext_vspace_above: 62, 102, 1733, 1733, 3762
__enumext_vspace_above_v: . 62, 1761, 1761, 3866
\l__enumext_vspace_above_v_skip .. 1763, 1767,
 1769

```

|                                                   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>\__enumext_vspace_above_vii:</code>         | <a href="#">63</a> , <a href="#">120</a> , <a href="#">1783</a> , <a href="#">1783</a> , <a href="#">4691</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <code>\l__enumext_vspace_above_vii_skip</code>    | <a href="#">1785</a> , <a href="#">1789</a> , <a href="#">1791</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <code>\__enumext_vspace_above_viii:</code>        | <a href="#">63</a> , <a href="#">1783</a> , <a href="#">1794</a> , <a href="#">4958</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <code>\l__enumext_vspace_above_viii_skip</code>   | <a href="#">1796</a> , <a href="#">1800</a> , <a href="#">1802</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <code>\l__enumext_vspace_b_star_v_bool</code>     | <a href="#">1776</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>\l__enumext_vspace_b_star_vii_bool</code>   | <a href="#">1809</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>\l__enumext_vspace_b_star_viii_bool</code>  | <a href="#">1820</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>\l__enumext_vspace_b_star_X_bool</code>     | <a href="#">96</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <code>\__enumext_vspace_below:</code>             | <a href="#">62</a> , <a href="#">103</a> , <a href="#">1747</a> , <a href="#">1747</a> , <a href="#">3800</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <code>\__enumext_vspace_below_v:</code>           | <a href="#">63</a> , <a href="#">1772</a> , <a href="#">1772</a> , <a href="#">3939</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <code>\l__enumext_vspace_below_v_skip</code>      | <a href="#">1774</a> , <a href="#">1778</a> , <a href="#">1780</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <code>\__enumext_vspace_below_vii:</code>         | <a href="#">63</a> , <a href="#">120</a> , <a href="#">1805</a> , <a href="#">1805</a> , <a href="#">4701</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <code>\l__enumext_vspace_below_vii_skip</code>    | <a href="#">1807</a> , <a href="#">1811</a> , <a href="#">1813</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <code>\__enumext_vspace_below_viii:</code>        | <a href="#">63</a> , <a href="#">1805</a> , <a href="#">1816</a> , <a href="#">4966</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <code>\l__enumext_vspace_below_viii_skip</code>   | <a href="#">1818</a> , <a href="#">1822</a> , <a href="#">1824</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <code>\__enumext_widest_from:nNNn</code>          | <a href="#">46</a> , <a href="#">889</a> , <a href="#">889</a> , <a href="#">904</a> , <a href="#">923</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <code>\g__enumext_widest_label_tl</code>          | <a href="#">28</a> , <a href="#">41</a> , <a href="#">63</a> , <a href="#">622</a> , <a href="#">626</a> , <a href="#">630</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <code>\l__enumext_wrap_label_opt_v_bool</code>    | <a href="#">3447</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>\l__enumext_wrap_label_opt_vii_bool</code>  | <a href="#">121</a> , <a href="#">4758</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <code>\l__enumext_wrap_label_opt_viii_bool</code> | <a href="#">127</a> , <a href="#">5002</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <code>\l__enumext_wrap_label_opt_X_bool</code>    | <a href="#">96</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <code>\l__enumext_wrap_label_v_bool</code>        | <a href="#">3443</a> , <a href="#">3447</a> , <a href="#">3455</a> , <a href="#">3502</a> , <a href="#">3510</a> , <a href="#">4251</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <code>\l__enumext_wrap_label_vii_bool</code>      | <a href="#">121</a> , <a href="#">4758</a> , <a href="#">4762</a> , <a href="#">4770</a> , <a href="#">4834</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| <code>\l__enumext_wrap_label_viii_bool</code>     | <a href="#">127</a> , <a href="#">5002</a> , <a href="#">5006</a> , <a href="#">5015</a> , <a href="#">5074</a> , <a href="#">5082</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| <code>\l__enumext_wrap_label_X_bool</code>        | <a href="#">96</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <code>\__enumext_wrapper_label_v:n</code>         | <a href="#">3508</a> , <a href="#">3512</a> , <a href="#">4271</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <code>\__enumext_wrapper_label_vii:n</code>       | <a href="#">4836</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>\__enumext_wrapper_label_viii:n</code>      | <a href="#">5080</a> , <a href="#">5084</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <code>\l__enumext_write_aux_file_tl.</code>       | <a href="#">31</a> , <a href="#">79</a> , <a href="#">89</a> , <a href="#">161</a> , <a href="#">2590</a> , <a href="#">2596</a> , <a href="#">3108</a> , <a href="#">3114</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <code>enumext*</code>                             | <a href="#">5</a> , <a href="#">4638</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>enumXi</code>                               | <a href="#">585</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <code>enumXii</code>                              | <a href="#">585</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <code>enumXiii</code>                             | <a href="#">585</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <code>enumXiv</code>                              | <a href="#">585</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <code>enumXv</code>                               | <a href="#">585</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <code>enumXvi</code>                              | <a href="#">585</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <code>enumXvii</code>                             | <a href="#">585</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <code>enumXviii</code>                            | <a href="#">585</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| Environments provide by <code>enumext</code> :    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| <code>anskey*</code>                              | <a href="#">30</a> , <a href="#">69</a> , <a href="#">75</a> , <a href="#">78–81</a> , <a href="#">83</a> , <a href="#">84</a> , <a href="#">86</a> , <a href="#">87</a> , <a href="#">101</a> , <a href="#">120</a> , <a href="#">130</a> , <a href="#">131</a> , <a href="#">136</a> , <a href="#">138</a>                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <code>enumext*</code>                             | <a href="#">27</a> , <a href="#">28</a> , <a href="#">31–33</a> , <a href="#">37–41</a> , <a href="#">43</a> , <a href="#">44</a> , <a href="#">46–50</a> , <a href="#">52</a> , <a href="#">59</a> , <a href="#">60</a> , <a href="#">63–66</a> , <a href="#">68–71</a> , <a href="#">73–83</a> , <a href="#">86</a> , <a href="#">88</a> , <a href="#">89</a> , <a href="#">94</a> , <a href="#">95</a> , <a href="#">100</a> , <a href="#">101</a> , <a href="#">106</a> , <a href="#">107</a> , <a href="#">114</a> , <a href="#">115</a> , <a href="#">117</a> , <a href="#">118</a> , <a href="#">120</a> , <a href="#">122–126</a> , <a href="#">128–133</a> , <a href="#">137</a> , <a href="#">140</a> , <a href="#">141</a> |
| <code>enumext</code>                              | <a href="#">27</a> , <a href="#">28</a> , <a href="#">32</a> , <a href="#">33</a> , <a href="#">37–41</a> , <a href="#">43–48</a> , <a href="#">50–55</a> , <a href="#">57</a> , <a href="#">60–62</a> , <a href="#">64–66</a> , <a href="#">68–71</a> , <a href="#">73–83</a> , <a href="#">86</a> , <a href="#">88</a> , <a href="#">89</a> , <a href="#">92–95</a> , <a href="#">98</a> , <a href="#">101</a> , <a href="#">104</a> , <a href="#">105</a> , <a href="#">109</a> , <a href="#">114</a> , <a href="#">117</a> , <a href="#">119</a> , <a href="#">120</a> , <a href="#">122</a> , <a href="#">125</a> , <a href="#">130–133</a> , <a href="#">137</a> , <a href="#">138</a> , <a href="#">140</a>                    |

|                                   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-----------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>keyans*</code>              | <a href="#">27</a> , <a href="#">28</a> , <a href="#">30–34</a> , <a href="#">37–40</a> , <a href="#">43–50</a> , <a href="#">52</a> , <a href="#">59</a> , <a href="#">60</a> , <a href="#">63</a> , <a href="#">69</a> , <a href="#">70</a> , <a href="#">73</a> , <a href="#">74</a> , <a href="#">76</a> , <a href="#">84</a> , <a href="#">88</a> , <a href="#">94</a> , <a href="#">100</a> , <a href="#">106</a> , <a href="#">107</a> , <a href="#">115</a> , <a href="#">116</a> , <a href="#">124</a> , <a href="#">125</a> , <a href="#">137</a> , <a href="#">139</a> , <a href="#">141</a>                                                                                                                                                                                          |
| <code>keyanspic</code>            | <a href="#">27</a> , <a href="#">28</a> , <a href="#">30</a> , <a href="#">31</a> , <a href="#">34</a> , <a href="#">40</a> , <a href="#">45</a> , <a href="#">69</a> , <a href="#">70</a> , <a href="#">73</a> , <a href="#">76</a> , <a href="#">84</a> , <a href="#">88–90</a> , <a href="#">94</a> , <a href="#">106–111</a> , <a href="#">113</a> , <a href="#">139</a>                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <code>keyans</code>               | <a href="#">27</a> , <a href="#">28</a> , <a href="#">30</a> , <a href="#">31</a> , <a href="#">33</a> , <a href="#">34</a> , <a href="#">37</a> , <a href="#">38</a> , <a href="#">40</a> , <a href="#">41</a> , <a href="#">45–48</a> , <a href="#">50</a> , <a href="#">52</a> , <a href="#">54</a> , <a href="#">57</a> , <a href="#">60–63</a> , <a href="#">69</a> , <a href="#">70</a> , <a href="#">73</a> , <a href="#">74</a> , <a href="#">76</a> , <a href="#">84</a> , <a href="#">88–90</a> , <a href="#">94</a> , <a href="#">96–98</a> , <a href="#">104–106</a> , <a href="#">108–110</a> , <a href="#">112</a> , <a href="#">117</a> , <a href="#">126</a> , <a href="#">137</a> , <a href="#">139</a>                                                                         |
| Environments:                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>center</code>               | <a href="#">114</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>description</code>          | <a href="#">94</a> , <a href="#">114</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <code>enumerate</code>            | <a href="#">114</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>flushleft</code>            | <a href="#">114</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>flushright</code>           | <a href="#">114</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>itemize</code>              | <a href="#">114</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>list</code>                 | <a href="#">32</a> , <a href="#">35</a> , <a href="#">36</a> , <a href="#">81</a> , <a href="#">94</a> , <a href="#">98</a> , <a href="#">102</a> , <a href="#">104</a> , <a href="#">106</a> , <a href="#">108–110</a> , <a href="#">114</a> , <a href="#">117</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>lrbox</code>                | <a href="#">123</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>minipage</code>             | <a href="#">32</a> , <a href="#">35</a> , <a href="#">36</a> , <a href="#">38</a> , <a href="#">39</a> , <a href="#">52</a> , <a href="#">54–56</a> , <a href="#">108</a> , <a href="#">109</a> , <a href="#">111</a> , <a href="#">113</a> , <a href="#">114</a> , <a href="#">117</a> , <a href="#">123</a> , <a href="#">124</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>multicols</code>            | <a href="#">52–56</a> , <a href="#">60</a> , <a href="#">102</a> , <a href="#">103</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <code>quotation</code>            | <a href="#">114</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>quote</code>                | <a href="#">114</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>scontents</code>            | <a href="#">84</a> , <a href="#">86</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <code>tabbing</code>              | <a href="#">114</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>trivlist</code>             | <a href="#">114</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>verbatim</code>             | <a href="#">114</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>verse</code>                | <a href="#">114</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| exp commands:                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>\exp_after:wN</code>        | <a href="#">5183</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <code>\exp_args:Ne</code>         | <a href="#">2996</a> , <a href="#">3004</a> , <a href="#">3671</a> , <a href="#">5171</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <code>\exp_args:NV</code>         | <a href="#">2716</a> , <a href="#">2871</a> , <a href="#">3387</a> , <a href="#">3405</a> , <a href="#">3427</a> , <a href="#">5456</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <code>\exp_not:N</code>           | <a href="#">54</a> , <a href="#">625</a> , <a href="#">752</a> , <a href="#">796</a> , <a href="#">812</a> , <a href="#">864</a> , <a href="#">1060</a> , <a href="#">1063</a> , <a href="#">1074</a> , <a href="#">1075</a> , <a href="#">1076</a> , <a href="#">1087</a> , <a href="#">1088</a> , <a href="#">1099</a> , <a href="#">1100</a> , <a href="#">2655</a> , <a href="#">2687</a> , <a href="#">2688</a> , <a href="#">3147</a> , <a href="#">3212</a> , <a href="#">3213</a> , <a href="#">3225</a> , <a href="#">3226</a> , <a href="#">5059</a> , <a href="#">5060</a> , <a href="#">5180</a>                                                                                                                                                                                     |
| <code>\exp_not:n</code>           | <a href="#">287</a> , <a href="#">302</a> , <a href="#">315</a> , <a href="#">323</a> , <a href="#">331</a> , <a href="#">691</a> , <a href="#">711</a> , <a href="#">752</a> , <a href="#">796</a> , <a href="#">812</a> , <a href="#">864</a> , <a href="#">1061</a> , <a href="#">1854</a> , <a href="#">1863</a> , <a href="#">2316</a> , <a href="#">2419</a> , <a href="#">2431</a> , <a href="#">2593</a> , <a href="#">2621</a> , <a href="#">2631</a> , <a href="#">2641</a> , <a href="#">2655</a> , <a href="#">2656</a> , <a href="#">2963</a> , <a href="#">2976</a> , <a href="#">2986</a> , <a href="#">3111</a> , <a href="#">3149</a> , <a href="#">3151</a> , <a href="#">4083</a> , <a href="#">5285</a> , <a href="#">5295</a> , <a href="#">5488</a> , <a href="#">5493</a> |
| F                                 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>\fbox</code>                | <a href="#">2297</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <code>\fboxrule</code>            | <a href="#">2297</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <code>\fboxsep</code>             | <a href="#">2297</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| file commands:                    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>\file_input_stop:</code>    | <a href="#">5892</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <code>first</code>                | <a href="#">1113</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <code>font</code>                 | <a href="#">644</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>\footnote</code>            | <a href="#">37</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <code>\footnote</code>            | <a href="#">37</a> , <a href="#">442</a> , <a href="#">472</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <code>\footnotemark</code>        | <a href="#">452</a> , <a href="#">482</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <code>\footnotesize</code>        | <a href="#">2688</a> , <a href="#">3213</a> , <a href="#">3226</a> , <a href="#">5060</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <code>\footnotetext</code>        | <a href="#">438</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>\foreachkeyans</code>       | <a href="#">17</a> , <a href="#">134</a> , <a href="#">5421</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| G                                 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>\getkeyans</code>           | <a href="#">17</a> , <a href="#">130</a> , <a href="#">5169</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| group commands:                   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>\group_begin:</code>        | <a href="#">2686</a> , <a href="#">2731</a> , <a href="#">2906</a> , <a href="#">2993</a> , <a href="#">3211</a> , <a href="#">3224</a> , <a href="#">5058</a> , <a href="#">5226</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <code>\group_end:</code>          | <a href="#">2693</a> , <a href="#">2747</a> , <a href="#">3010</a> , <a href="#">3218</a> , <a href="#">3231</a> , <a href="#">5065</a> , <a href="#">5233</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| H                                 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>\hbadness</code>            | <a href="#">4869</a> , <a href="#">5140</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| hbox commands:                    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>\hbox_overlap_left:n</code> | <a href="#">3292</a> , <a href="#">4827</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |



|                       |                                                                                                                                                                                                                                                                                                                                                                                     |
|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| \hbox_set:Nn          | 614, 4129                                                                                                                                                                                                                                                                                                                                                                           |
| \hbox_set_end:        | 4868, 5139                                                                                                                                                                                                                                                                                                                                                                          |
| \hbox_set_to_wd:Nnw   | 4845, 5112                                                                                                                                                                                                                                                                                                                                                                          |
| \hfill                | 674, 679, 685, 686, 1672, 1699, 2655, 3147, 4541, 4605                                                                                                                                                                                                                                                                                                                              |
| hook commands:        |                                                                                                                                                                                                                                                                                                                                                                                     |
| \hook_gput_code:nnn   | 5, 205, 209, 213, 386                                                                                                                                                                                                                                                                                                                                                               |
| \hook_gremove_code:nn | 86, 2922                                                                                                                                                                                                                                                                                                                                                                            |
| \hook_gset_rule:nnnn  | 387                                                                                                                                                                                                                                                                                                                                                                                 |
| \hook_if_empty:nTF    | 2920                                                                                                                                                                                                                                                                                                                                                                                |
| \hyperlink            | 80, 90                                                                                                                                                                                                                                                                                                                                                                              |
| \hyperlink            | 2655, 3147                                                                                                                                                                                                                                                                                                                                                                          |
| \hypertarget          | 36                                                                                                                                                                                                                                                                                                                                                                                  |
| \hypertarget          | 413                                                                                                                                                                                                                                                                                                                                                                                 |
| <b>I</b>              |                                                                                                                                                                                                                                                                                                                                                                                     |
| \IfDocumentMetadataTF | 504, 520, 534, 547, 3309, 3484, 3998, 4006, 4014, 4050, 4058, 4066, 4152, 4161, 4169, 4176, 4181, 4229, 4238, 4329, 4337, 4539, 4603, 4649, 4657, 4803, 4911, 4919                                                                                                                                                                                                                  |
| \IfHyperBoolean       | 394                                                                                                                                                                                                                                                                                                                                                                                 |
| \IfPackageLoadedTF    | 7, 15, 390, 403                                                                                                                                                                                                                                                                                                                                                                     |
| \ignorespaces         | 1063, 1076, 1088, 1100, 4654, 4731, 4764, 4787, 4794, 4840, 4860, 4916, 4974, 5008, 5040, 5107, 5127                                                                                                                                                                                                                                                                                |
| \inputlineno          | 287, 302, 315, 323, 331                                                                                                                                                                                                                                                                                                                                                             |
| int commands:         |                                                                                                                                                                                                                                                                                                                                                                                     |
| \int_add:Nn           | 4449, 4498                                                                                                                                                                                                                                                                                                                                                                          |
| \int_case:nn          | 1237, 1362, 2162, 2188, 2227, 2251                                                                                                                                                                                                                                                                                                                                                  |
| \int_case:nnTF        | 241                                                                                                                                                                                                                                                                                                                                                                                 |
| \int_compare:nNnTF    | 568, 784, 800, 819, 826, 1332, 1351, 1505, 1523, 1635, 1654, 1666, 1694, 2275, 2281, 2755, 2759, 2763, 2771, 2817, 2821, 2825, 3022, 3043, 3084, 3089, 3094, 3119, 3207, 3653, 3664, 3686, 3699, 3715, 3730, 3745, 3786, 3851, 3855, 3883, 3908, 3924, 4101, 4212, 4216, 4419, 4429, 4445, 4468, 4478, 4494, 4667, 4671, 4709, 4719, 4871, 4883, 4932, 4944, 5142, 5154, 5339, 5471 |
| \int_compare_p:nNn    | 256, 266, 278, 279, 293, 294, 1641, 1642, 2168, 2194, 2538, 2548, 2560, 2561, 2576, 2617, 2794, 2795, 2806, 2807, 2959, 3504, 3696, 5076                                                                                                                                                                                                                                            |
| \int_decr:N           | 4448, 4497                                                                                                                                                                                                                                                                                                                                                                          |
| \int_eval:n           | 373, 918, 2437, 2588, 2688, 3102, 3213, 3226, 3587, 3632, 4437, 4486, 5060                                                                                                                                                                                                                                                                                                          |
| \int_from_alph:n      | 883, 897                                                                                                                                                                                                                                                                                                                                                                            |
| \int_from_roman:n     | 885, 899                                                                                                                                                                                                                                                                                                                                                                            |
| \int_gadd:Nn          | 4450, 4499                                                                                                                                                                                                                                                                                                                                                                          |
| \int_gdecr:N          | 2171, 2176, 2180, 2184, 2197                                                                                                                                                                                                                                                                                                                                                        |
| \int_gincr:N          | 2004, 2009, 2600, 3157, 3246, 3280, 3462, 3775, 3875, 4255, 4737, 4813, 4980, 5049                                                                                                                                                                                                                                                                                                  |
| \int_gset:Nn          | 450, 480, 2220                                                                                                                                                                                                                                                                                                                                                                      |
| \int_gset_eq:NN       | 447, 477, 1903, 1910, 1916, 1922, 1930, 1937, 1943, 1949                                                                                                                                                                                                                                                                                                                            |
| \int_gzero:N          | 344, 345, 346, 1680, 1707, 2287, 3015, 3791, 3929, 4894, 5166                                                                                                                                                                                                                                                                                                                       |
| \int_if_exist:NnTF    | 1878, 1914, 1920, 1941, 1947, 2125                                                                                                                                                                                                                                                                                                                                                  |
| \int_incr:N           | 2770, 3470, 3652, 3846, 4100, 4250, 4666, 4736, 4931, 4979, 5014                                                                                                                                                                                                                                                                                                                    |
| \int_mod:nn           | 4885, 5156                                                                                                                                                                                                                                                                                                                                                                          |
| \int_new:N            | 24, 25, 26, 27, 28, 29, 57, 58, 83, 100, 119, 140, 141, 152, 154, 155, 156, 158, 169, 170, 178, 179, 180, 181, 182, 1880, 2128                                                                                                                                                                                                                                                      |
| \int_set:Nn           | 879, 883, 885, 2017, 2024, 2036, 2045, 2907, 4323, 4324, 4364, 4395, 4418, 4424, 4440, 4467, 4473, 4489, 4869, 5140, 5335, 5473                                                                                                                                                                                                                                                     |
| \int_set_eq:NN        | 2005, 2010, 4447, 4496                                                                                                                                                                                                                                                                                                                                                              |
| \int_sign:n           | 2222                                                                                                                                                                                                                                                                                                                                                                                |

|                         |                                                                                                                                                                     |
|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| \int_step_function:nnN  | 2554, 2567, 2581                                                                                                                                                    |
| \int_step_function:nnnN | 5477                                                                                                                                                                |
| \int_step_inline:nn     | 5387                                                                                                                                                                |
| \int_step_inline:nnn    | 4325                                                                                                                                                                |
| \int_to_roman:n         | 217, 2534, 2571                                                                                                                                                     |
| \int_use:N              | 366, 371, 372, 1333, 1352, 1667, 2019, 2026, 2038, 2047, 3587, 3607, 3632, 3672, 3716, 3725, 3740, 3746, 4422, 4423, 4435, 4471, 4472, 4484, 5808, 5812, 5818, 5822 |
| \int_zero:N             | 3477, 4269, 4875, 4999, 5146                                                                                                                                        |
| \item                   | 91, 96, 121, 123, 126, 129, 379, 2455, 2461, 2486, 2492, 2614, 3121, 3124, 3298, 3466, 4156, 4157, 4651, 4653, 4913, 4915, 5047                                     |
| \item*                  | 5, 15, 73, 3464                                                                                                                                                     |
| item-pos*               | 3361                                                                                                                                                                |
| item-sym*               | 3361                                                                                                                                                                |
| \itemindent             | 99                                                                                                                                                                  |
| \itemindent             | 98                                                                                                                                                                  |
| itemindent              | 1007                                                                                                                                                                |
| \itemsep                | 4145                                                                                                                                                                |
| \itemwidth              | 584, 2297, 3806, 3812, 3943, 3949, 4458, 4462, 4507, 4511                                                                                                           |

K

|                                           |                                                                |
|-------------------------------------------|----------------------------------------------------------------|
| keyans                                    | 14, 3952                                                       |
| keyans*                                   | 14, 4901                                                       |
| keyanspic                                 | 15, 4147                                                       |
| Keys for \anskey provide by enumext:      |                                                                |
| break-col                                 | 80, 81, 84-86                                                  |
| item-join                                 | 80, 81, 84-86                                                  |
| item-pos*                                 | 80, 81, 84-86                                                  |
| item-star                                 | 80, 81, 84-86                                                  |
| item-sym*                                 | 80, 81, 84-86                                                  |
| Keys for anskey* provide by enumext:      |                                                                |
| break-col                                 | 80, 81, 84-86                                                  |
| item-join                                 | 80, 81, 84-86                                                  |
| item-pos*                                 | 80, 81, 84-86                                                  |
| item-star                                 | 80, 81, 84-86                                                  |
| item-sym*                                 | 80, 81, 84-86                                                  |
| Keys for environments provide by enumext: |                                                                |
| above*                                    | 29, 48, 61-63, 102, 120                                        |
| above                                     | 29, 48, 61-63, 102, 120, 125                                   |
| after                                     | 50, 51, 103, 120, 126                                          |
| align                                     | 29, 42, 91, 93, 97, 122, 136                                   |
| base-fix                                  | 47, 48, 64, 76, 100                                            |
| before*                                   | 50, 102, 120, 125                                              |
| before                                    | 50                                                             |
| below*                                    | 29, 61-63, 103, 120                                            |
| below                                     | 29, 61-63, 103, 120, 126                                       |
| check-ans                                 | 31, 32, 34, 68-70, 72, 73, 76, 87, 90, 103, 104, 120, 124, 138 |
| columns-sep                               | 52, 102, 124                                                   |
| columns                                   | 29, 52, 61, 102                                                |
| first                                     | 50, 51, 123                                                    |
| font                                      | 41, 93, 97, 112, 122                                           |
| item-pos*                                 | 92, 94                                                         |
| item-sym*                                 | 30, 92, 94                                                     |
| itemindent                                | 29, 48, 49, 92, 96, 123                                        |
| itemsep                                   | 47, 100, 124                                                   |
| label-pos                                 | 108, 110-112                                                   |
| label-sep                                 | 109                                                            |
| labelsep                                  | 41, 98, 122                                                    |
| labelwidth                                | 40, 41, 43, 45, 46, 98, 122                                    |
| label                                     | 28, 40-42, 45, 46, 109, 114                                    |
| layout-sep                                | 109                                                            |

|                      |                                                                                                                                                                                                              |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| layout-sty           | 109, 113                                                                                                                                                                                                     |
| layout-top           | 109                                                                                                                                                                                                          |
| lisparindent         | 100                                                                                                                                                                                                          |
| list-indent          | 29, 48, 49, 109                                                                                                                                                                                              |
| list-offset          | 48, 49, 103, 106                                                                                                                                                                                             |
| listparindent        | 48, 123                                                                                                                                                                                                      |
| mark-ans             | 73, 76, 81                                                                                                                                                                                                   |
| mark-pos             | 73, 74, 136                                                                                                                                                                                                  |
| mark-ref             | 73, 76, 78, 80                                                                                                                                                                                               |
| mini-env             | 29, 37-39, 52, 60, 61, 76, 103, 114, 117, 118, 120, 125                                                                                                                                                      |
| mini-right*          | 29, 32, 52, 76, 117, 118, 120                                                                                                                                                                                |
| mini-right           | 29, 32, 52, 60, 76, 117, 118, 120                                                                                                                                                                            |
| mini-sep             | 29, 52, 76, 103                                                                                                                                                                                              |
| mode-box             | 41, 92-94, 97                                                                                                                                                                                                |
| no-store             | 31, 68-71, 76, 82, 92                                                                                                                                                                                        |
| noitemsep            | 47                                                                                                                                                                                                           |
| nosep                | 47                                                                                                                                                                                                           |
| parindent            | 100                                                                                                                                                                                                          |
| parsep               | 47, 100, 109, 123                                                                                                                                                                                            |
| partopsep            | 47                                                                                                                                                                                                           |
| ref                  | 28, 33, 42, 43, 45, 137                                                                                                                                                                                      |
| resume*              | 28, 64, 68, 69, 76, 103, 120, 132                                                                                                                                                                            |
| resume               | 28, 35, 64-69, 76, 103, 120, 132                                                                                                                                                                             |
| rightmargin          | 48, 114                                                                                                                                                                                                      |
| save-ans             | 30, 35, 64-69, 71, 72, 74-76, 82-84, 87-89, 96, 104, 111, 122, 125-127, 130, 132, 137                                                                                                                        |
| save-key             | 30, 64, 75, 100, 120                                                                                                                                                                                         |
| save-pos             | 76                                                                                                                                                                                                           |
| save-ref             | 31, 37, 73, 76, 78-80, 88, 90, 96, 127                                                                                                                                                                       |
| save-sep             | 73, 74, 76, 88, 127                                                                                                                                                                                          |
| series               | 28, 64-68, 76, 100, 103, 120, 132                                                                                                                                                                            |
| show-ans             | 73, 74, 76, 78, 79, 81, 96, 112, 127                                                                                                                                                                         |
| show-length          | 33, 50, 137                                                                                                                                                                                                  |
| show-pos             | 30, 73, 74, 78, 79, 81, 90, 96, 112, 127                                                                                                                                                                     |
| start*               | 29, 45, 46, 64                                                                                                                                                                                               |
| start                | 29, 32, 45, 46, 64                                                                                                                                                                                           |
| store-key            | 75                                                                                                                                                                                                           |
| topsep               | 47, 48, 109                                                                                                                                                                                                  |
| widest               | 28, 32, 46                                                                                                                                                                                                   |
| wrap-ans             | 40, 73, 76, 78, 80                                                                                                                                                                                           |
| wrap-key             | 31, 73, 74, 97, 112                                                                                                                                                                                          |
| wrap-label*          | 29, 41, 92, 93, 96, 97, 121, 122, 127                                                                                                                                                                        |
| wrap-label           | 29, 41, 92, 93, 96, 97, 109, 112, 121, 122, 127                                                                                                                                                              |
| wrap-opt             | 73, 74, 76, 96, 112                                                                                                                                                                                          |
| keys commands:       |                                                                                                                                                                                                              |
| \keys_define:nn      | 636, 646, 668, 703, 719, 765, 834, 907, 933, 975, 1009, 1032, 1106, 1115, 1194, 1211, 1718, 1829, 2072, 2133, 2292, 2336, 2378, 2383, 2698, 2849, 2885, 3363, 3379, 3399, 3419, 4072, 5197, 5297, 5413, 5421 |
| \keys_if_exist_p:nn  | 5409, 5410                                                                                                                                                                                                   |
| \l_keys_key_str      | 82, 85, 2716, 2871, 3387, 3405, 3427, 5456, 5564                                                                                                                                                             |
| \keys_precompile:nnN | 131, 201, 201, 5199, 5203, 5207, 5211, 5215, 5219, 5439                                                                                                                                                      |
| \keys_set:nn         | 660, 1001, 1217, 1723, 1728, 1966, 1971, 2058, 2066, 2736, 3666, 3671, 3862, 4090, 4093, 4111, 4684, 4953, 5301, 5306, 5307, 5308, 5309, 5312, 5317, 5318, 5319, 5320, 5321, 5322, 5323, 5355, 5465          |
| \keys_set_known:nn   | 3003                                                                                                                                                                                                         |
| keyval commands:     |                                                                                                                                                                                                              |
| \keyval_parse:NNn    | 1843, 2408, 5273                                                                                                                                                                                             |
| L                    |                                                                                                                                                                                                              |
| label                | 717, 763, 834                                                                                                                                                                                                |

|                            |                                                                                                                                                                                                        |
|----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| label-pos                  | 4072                                                                                                                                                                                                   |
| label-sep                  | 4072                                                                                                                                                                                                   |
| Labels provide by enumext: |                                                                                                                                                                                                        |
| \Alph*                     | 40, 41                                                                                                                                                                                                 |
| \Roman*                    | 40, 41                                                                                                                                                                                                 |
| \alph*                     | 40, 41                                                                                                                                                                                                 |
| \arabic*                   | 33, 40, 41                                                                                                                                                                                             |
| \roman*                    | 40, 41                                                                                                                                                                                                 |
| \labelsep                  | 4141                                                                                                                                                                                                   |
| labelsep                   | 644                                                                                                                                                                                                    |
| \labelwidth                | 41                                                                                                                                                                                                     |
| \labelwidth                | 4141                                                                                                                                                                                                   |
| labelwidth                 | 644                                                                                                                                                                                                    |
| \lastnodetype              | 241                                                                                                                                                                                                    |
| layout-sep                 | 4072                                                                                                                                                                                                   |
| layout-sty                 | 4072                                                                                                                                                                                                   |
| layout-top                 | 4072                                                                                                                                                                                                   |
| \leftmargin                | 99                                                                                                                                                                                                     |
| \leftmargin                | 98, 4141                                                                                                                                                                                               |
| legacy commands:           |                                                                                                                                                                                                        |
| \legacy_if:nTF             | 4798, 4801, 5090, 5093                                                                                                                                                                                 |
| \legacy_if_gset_false:n    | 574, 4554                                                                                                                                                                                              |
| \legacy_if_set_false:n     | 4800, 5092                                                                                                                                                                                             |
| \legacy_if_set_true:n      | 4763, 4786, 4793, 4807, 5007, 5039                                                                                                                                                                     |
| \linewidth                 | 103                                                                                                                                                                                                    |
| \linewidth                 | 3770, 3806, 3872, 3943, 4322, 4367, 4398, 4520, 4585                                                                                                                                                   |
| \list                      | 377                                                                                                                                                                                                    |
| list-indent                | 1007                                                                                                                                                                                                   |
| list-offset                | 1007                                                                                                                                                                                                   |
| \listparindent             | 4143                                                                                                                                                                                                   |
| listparindent              | 1007                                                                                                                                                                                                   |
| M                          |                                                                                                                                                                                                        |
| \makebox                   | 114                                                                                                                                                                                                    |
| \makebox                   | 2513, 2515, 3345, 3532, 4246, 4831, 5102                                                                                                                                                               |
| \makelabel                 | 91, 93, 97, 114                                                                                                                                                                                        |
| \makelabel                 | 91, 96, 3325, 3341, 3518, 3528                                                                                                                                                                         |
| mark-ans                   | 2290                                                                                                                                                                                                   |
| mark-pos                   | 2290, 2334                                                                                                                                                                                             |
| mark-ref                   | 2290                                                                                                                                                                                                   |
| mini-env                   | 1192                                                                                                                                                                                                   |
| mini-sep                   | 1192                                                                                                                                                                                                   |
| \minipage                  | 383                                                                                                                                                                                                    |
| \miniright                 | 11, 60, 1633, 1684, 1711, 3789, 3927                                                                                                                                                                   |
| mode commands:             |                                                                                                                                                                                                        |
| \mode_if_math:TF           | 2779, 2833                                                                                                                                                                                             |
| \mode_if_vertical:TF       | 1271, 1299, 1319, 1343, 1494, 1515                                                                                                                                                                     |
| \mode_leave_vertical:      | 990, 997, 1060, 1074, 2511, 3290, 4825                                                                                                                                                                 |
| mode-box                   | 634                                                                                                                                                                                                    |
| mmsg commands:             |                                                                                                                                                                                                        |
| \msg_error:nn              | 1686, 1713, 2740, 2773, 2777, 2831, 2939, 3853, 3857, 4103, 4159, 4214, 4669, 4934, 4946, 5324, 5383                                                                                                   |
| \msg_error:nnn             | 742, 788, 804, 856, 1637, 1644, 1651, 1682, 1709, 1978, 1982, 2097, 2722, 2781, 2799, 2811, 2819, 2823, 2827, 2835, 2877, 3393, 3411, 3433, 4673, 4939, 5185, 5194, 5266, 5371, 5402, 5411, 5448, 5469 |
| \msg_error:nnnn            | 2725, 2753, 2757, 2761, 2765, 2880, 3396, 3414, 3436, 3844, 4210, 4218, 4929, 5245, 5451                                                                                                               |
| \msg_error:nnnnn           | 690, 710, 2315, 4082                                                                                                                                                                                   |
| \msg_fatal:nn              | 3654                                                                                                                                                                                                   |



|                         |                                                                                                                                                                                                                                                                                                                                                      |                                     |                                                                                                                                          |
|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| \msg_fatal:nnn          | 588                                                                                                                                                                                                                                                                                                                                                  | \par                                | 1282, 1310, 1348, 1520, 1624, 1630, 1669, 1674, 1696, 1701, 2663, 3753, 3914, 3932, 4192, 4195, 4342, 4556, 4571, 4617, 4631, 4874, 5145 |
| \msg_info:nnn           | 9, 12, 17, 20, 392, 405                                                                                                                                                                                                                                                                                                                              | para commands:                      |                                                                                                                                          |
| \msg_line_context:      | 5529, 5534, 5539, 5568, 5573, 5578, 5593, 5608, 5612, 5616, 5620, 5624, 5628, 5635, 5642, 5648, 5662, 5666, 5671, 5675, 5679, 5683, 5688, 5692, 5696, 5700, 5705, 5740, 5744, 5749, 5754, 5758, 5763, 5839, 5843, 5848, 5853, 5858, 5862, 5866, 5870, 5874, 5878, 5882, 5886, 5890                                                                   | \para_end:                          | 4891, 5163                                                                                                                               |
| \msg_log:nnn            | 2117, 2122, 2127                                                                                                                                                                                                                                                                                                                                     | \parbox                             | 2297                                                                                                                                     |
| \msg_log:nnnnn          | 370, 2260, 2265, 2270                                                                                                                                                                                                                                                                                                                                | \parindent                          | 4856, 5123                                                                                                                               |
| \msg_log:nnnnnn         | 362                                                                                                                                                                                                                                                                                                                                                  | \parsep                             | 53, 109                                                                                                                                  |
| \msg_new:nnn            | 5496, 5500, 5504, 5508, 5513, 5526, 5531, 5536, 5541, 5550, 5558, 5562, 5566, 5571, 5576, 5591, 5606, 5610, 5614, 5618, 5622, 5626, 5630, 5639, 5645, 5651, 5655, 5659, 5664, 5669, 5673, 5677, 5681, 5686, 5690, 5694, 5698, 5703, 5738, 5742, 5747, 5752, 5756, 5761, 5837, 5841, 5846, 5851, 5856, 5860, 5864, 5868, 5872, 5876, 5880, 5884, 5888 | \parsep                             | 991, 3629, 4125, 4134                                                                                                                    |
| \msg_new:nnnn           | 5517, 5708, 5717, 5726, 5732, 5765, 5775, 5785, 5795, 5805, 5815, 5825, 5831                                                                                                                                                                                                                                                                         | parsep                              | 931                                                                                                                                      |
| \msg_term:nnnn          | 2081, 2086, 3596, 3606, 3638, 3643                                                                                                                                                                                                                                                                                                                   | \parskip                            | 4857, 5124                                                                                                                               |
| \msg_term:nnnnn         | 2241                                                                                                                                                                                                                                                                                                                                                 | \partopsep                          | 3630, 3930, 4144                                                                                                                         |
| \msg_warning:nn         | 3788, 3926                                                                                                                                                                                                                                                                                                                                           | partopsep                           | 931                                                                                                                                      |
| \msg_warning:nnnn       | 2278, 2284, 3544, 3549, 4421, 4434, 4470, 4483                                                                                                                                                                                                                                                                                                       | peek commands:                      |                                                                                                                                          |
| \msg_warning:nnnnn      | 2236, 2246                                                                                                                                                                                                                                                                                                                                           | \peek_meaning:NTF                   | 4742, 4756, 4771, 4782, 4985, 5000, 5016                                                                                                 |
| \multicolsep            | 102                                                                                                                                                                                                                                                                                                                                                  | \peek_meaning_remove:NTF            | 4749, 4992                                                                                                                               |
| \multicolsep            | 1336, 1508, 3736, 3899                                                                                                                                                                                                                                                                                                                               | \peek_remove_spaces:n               | 3471                                                                                                                                     |
| N                       |                                                                                                                                                                                                                                                                                                                                                      |                                     |                                                                                                                                          |
| \NeedsTeXFormat         | 3                                                                                                                                                                                                                                                                                                                                                    | \phantomsection                     | 36                                                                                                                                       |
| \NewCommandCopy         | 379                                                                                                                                                                                                                                                                                                                                                  | \phantomsection                     | 414                                                                                                                                      |
| \newcounter             | 591                                                                                                                                                                                                                                                                                                                                                  | prg commands:                       |                                                                                                                                          |
| \NewDocumentCommand     | 1633, 2728, 4206, 5169, 5224, 5331, 5380, 5458                                                                                                                                                                                                                                                                                                       | \prg_do_nothing:                    | 418                                                                                                                                      |
| \NewDocumentEnvironment | 3818, 3952, 4147, 4638, 4901                                                                                                                                                                                                                                                                                                                         | \prg_new_protected_conditional:Npnn | 219                                                                                                                                      |
| \newenvsc               | 2842                                                                                                                                                                                                                                                                                                                                                 | \prg_replicate:nn                   | 236                                                                                                                                      |
| \newlabel               | 37                                                                                                                                                                                                                                                                                                                                                   | \prg_return_false:                  | 223                                                                                                                                      |
| \newlabel               | 425                                                                                                                                                                                                                                                                                                                                                  | \prg_return_true:                   | 222                                                                                                                                      |
| no-store                | 2131                                                                                                                                                                                                                                                                                                                                                 | \printkeyans                        | 18, 130, 5224                                                                                                                            |
| \noindent               | 3777, 4529, 4594, 4874, 5145                                                                                                                                                                                                                                                                                                                         | prop commands:                      |                                                                                                                                          |
| \nointerlineskip        | 1345, 1348, 1517, 1520, 1674, 1701, 4529, 4594                                                                                                                                                                                                                                                                                                       | \prop_const_from_keyval:Nn          | 5372                                                                                                                                     |
| noitemsep               | 931                                                                                                                                                                                                                                                                                                                                                  | \prop_count:N                       | 364, 2437, 2588, 2690, 3102, 3215, 3228, 5062, 5474                                                                                      |
| \nopagebreak            | 1282, 1310, 1345, 1348, 1517, 1520, 1624, 1630                                                                                                                                                                                                                                                                                                       | \prop_get:NnNTF                     | 5398                                                                                                                                     |
| \normalfont             | 2687, 3212, 3225, 5059                                                                                                                                                                                                                                                                                                                               | \prop_gput_if_not_in:Nnn            | 2435                                                                                                                                     |
| nosep                   | 931                                                                                                                                                                                                                                                                                                                                                  | \prop_if_exist:NTF                  | 2115, 5189, 5467                                                                                                                         |
| P                       |                                                                                                                                                                                                                                                                                                                                                      |                                     |                                                                                                                                          |
| Packages:               |                                                                                                                                                                                                                                                                                                                                                      | \prop_item:Nn                       | 5191, 5491                                                                                                                               |
| caption                 | 117                                                                                                                                                                                                                                                                                                                                                  | \prop_new:N                         | 2118                                                                                                                                     |
| enumext                 | 27, 40, 42, 68, 94, 98, 108, 136                                                                                                                                                                                                                                                                                                                     | \ProvidesExplPackage                | 4                                                                                                                                        |
| enumitem                | 40                                                                                                                                                                                                                                                                                                                                                   | R                                   |                                                                                                                                          |
| expl3                   | 114                                                                                                                                                                                                                                                                                                                                                  | \raggedcolumns                      | 3739, 3902                                                                                                                               |
| footnotehyper           | 36, 38, 39                                                                                                                                                                                                                                                                                                                                           | \raisebox                           | 4284                                                                                                                                     |
| hyperref                | 31, 32, 36, 37, 80, 90, 122, 136                                                                                                                                                                                                                                                                                                                     | \ref                                | 78, 88                                                                                                                                   |
| latex-lab-block         | 36                                                                                                                                                                                                                                                                                                                                                   | ref                                 | 717, 763, 834                                                                                                                            |
| ltxcmd                  | 36                                                                                                                                                                                                                                                                                                                                                   | \refstepcounter                     | 4810, 5095                                                                                                                               |
| ltsockets               | 106                                                                                                                                                                                                                                                                                                                                                  | regex commands:                     |                                                                                                                                          |
| lua-visual-debug        | 55                                                                                                                                                                                                                                                                                                                                                   | \regex_match:nnTF                   | 221, 882, 884, 896, 898, 2935                                                                                                            |
| multicol                | 27, 136                                                                                                                                                                                                                                                                                                                                              | \regex_replace_once:nnN             | 229                                                                                                                                      |
| scontents               | 27, 83, 84                                                                                                                                                                                                                                                                                                                                           | \renewcommand                       | 752, 796, 812, 864                                                                                                                       |
| shortlst                | 114, 119, 123                                                                                                                                                                                                                                                                                                                                        | \RenewDocumentCommand               | 442, 472, 1684, 1711, 3298, 3325, 3341, 3466, 3518, 3528, 4157                                                                           |
| tagpdf                  | 106                                                                                                                                                                                                                                                                                                                                                  | \RequirePackage                     | 13, 21                                                                                                                                   |
| S                       |                                                                                                                                                                                                                                                                                                                                                      |                                     |                                                                                                                                          |
| \s                      | 2936                                                                                                                                                                                                                                                                                                                                                 | resume                              | 1827                                                                                                                                     |
| save-ans                | 2070                                                                                                                                                                                                                                                                                                                                                 | resume*                             | 1827                                                                                                                                     |
| save-key                | 2376                                                                                                                                                                                                                                                                                                                                                 | rightmargin                         | 1007                                                                                                                                     |
| save-ref                | 2290                                                                                                                                                                                                                                                                                                                                                 | \Roman                              | 41, 45, 46                                                                                                                               |
| save-sep                | 2290, 2334                                                                                                                                                                                                                                                                                                                                           | \Roman                              | 610                                                                                                                                      |
|                         |                                                                                                                                                                                                                                                                                                                                                      | \roman                              | 41, 45, 46                                                                                                                               |
|                         |                                                                                                                                                                                                                                                                                                                                                      | \roman                              | 611, 735, 5214                                                                                                                           |

## scan commands:

\scan\_stop: . . . . . 4156, 4651, 4913, 5180, 5183

## scontents internal commands:

\l\_\_scontents\_fname\_out\_tl . . . . . 2895

\\_\_scontents\_parse\_environment\_keys:n . 2901

\\_\_scontents\_rescan\_tokens:n . . . . . 2908

\l\_\_scontents\_storing\_bool . . . . . 2893

\l\_\_scontents\_writing\_bool . . . . . 2894

## seq commands:

\seq\_clear:N . . . . . 5333, 5476

\seq\_const\_from\_clist:Nn . . . . . 5326

\seq\_count:N . . . . . 365, 4348, 5337

\seq\_gclear:N . . . . . 467, 468, 497, 498

\seq\_gput\_right:Nn . . . . . 453, 454, 483, 484, 2444

\seq\_if\_empty:NTF . . . . . 460, 490, 5239, 5351

\seq\_if\_exist:NTF . . . . . 2120, 5237

\seq\_if\_in:NnTF . . . . . 5243

\seq\_item:Nn . . . . . 2933, 4335

\seq\_map\_function:NN . . . . . 5342

\seq\_map\_inline:Nn . . . . . 5252, 5260, 5352, 5353

\seq\_map\_pairwise\_function:NNN . . . . 462, 492

\seq\_new:N 120, 121, 123, 138, 171, 172, 173, 174, 2123

\seq\_pop\_left:NN . . . . . 5341

\seq\_put\_right:Nn . . . . . 4220, 5349, 5365, 5486

\seq\_set\_from\_clist:Nn . . . . . 5334

\seq\_set\_map\_e:NNn . . . . . 5343

\seq\_use:Nn . . . . . 201, 202, 5482

## series . . . . . 1827

\setcounter . . . . . 893, 897, 899, 3587, 3632, 4189

\setenumext . . . . . 6, 132, 5331

\setenumextmeta . . . . . 6, 133, 5372

show-ans . . . . . 2290, 2334

show-length . . . . . 1104

show-pos . . . . . 2334

## skip commands:

\skip\_add:Nn 1242, 1251, 1260, 1273, 1277, 1301, 1305,

1321, 1379, 1381, 1395, 1398, 1419, 1421, 1435, 1438,

1458, 1460, 1474, 1477, 1496, 1545, 1546, 1557, 1559,

4134, 4142

\skip\_gset:Nn . . . . . 1572, 1576, 1580

\skip\_gzero\_new:N . . . . . 1567, 1568

\skip\_horizontal:N . . 1075, 1087, 1099, 4828, 4840,

4878, 5107, 5149

\skip\_horizontal:n . . 1061, 2512, 2520, 3291, 3293,

4727, 4826, 4860, 4970, 5127

\skip\_if\_eq:nnTF 1240, 1249, 1258, 1365, 1405, 1445,

1533, 1569, 1591, 1735, 1749, 1763, 1774, 1785, 1796,

1807, 1818

\skip\_new:N 77, 78, 79, 84, 85, 86, 87, 88, 89, 144, 193

\skip\_set:Nn 1225, 1229, 1287, 1291, 1315, 1368, 1369,

1387, 1408, 1409, 1427, 1447, 1448, 1466, 1490, 1536,

1537, 1551, 1571, 1575, 1593, 1597, 1601, 1607, 1611,

1615, 4118

\skip\_set\_eq:NN 1326, 1327, 1329, 1336, 1501, 1502,

1503, 1508, 3585, 3628, 3629, 4857, 5124

\skip\_sub:Nn 1375, 1377, 1391, 1393, 1415, 1417, 1431,

1433, 1454, 1456, 1470, 1472, 1543, 1544, 1555, 1556

\skip\_use:N 1227, 1231, 1275, 1279, 1283, 1303, 1307,

1317, 1323, 1736, 1740, 1743, 1750, 1754, 1757, 3753

\skip\_vertical:N . 575, 578, 999, 4555, 4569, 4893,

5165

\skip\_vertical:n . . . . . 998, 4892, 5164

\skip\_zero:N 1335, 1349, 1487, 1488, 1489, 1507, 1521,

3630, 3736, 3899, 4144, 4145

\skip\_zero\_new:N . . . . . 1566, 1588, 1589, 1590

\c\_zero\_skip . 575, 578, 999, 1240, 1249, 1258, 1406,

1445, 1569, 1591, 1736, 1750, 1763, 1774, 1785, 1796,

1807, 1818, 4555, 4569, 4893, 5165

\small . . . . . 5202, 5206, 5210, 5214, 5218, 5222

\smash . . . . . 3343, 3530

## socket commands:

\socket\_assign\_plug:nn . . 4000, 4008, 4016, 4052,

4060, 4068

\socket\_new:nn . . . . . 3970, 4020

\socket\_new\_plug:nnn 3971, 3979, 3987, 4021, 4029,

4038

\socket\_use:n . . . . . 4053, 4061, 4069

\socket\_use:nn . . . . . 4001, 4009, 4017

start . . . . . 905

start\* . . . . . 905

start-list-tags . . . . . 3970, 4020

\stepcounter . . . . . 446, 476, 4128, 4277

stop-list-tags . . . . . 3970, 4020

stop-start-tags . . . . . 3970, 4020

## str commands:

\c\_backslash\_str 2781, 5529, 5534, 5539, 5544, 5546,

5548, 5553, 5555, 5653, 5657, 5661, 5671, 5675, 5683,

5684, 5688, 5700, 5701, 5705, 5706, 5727, 5729, 5733,

5735, 5763, 5826, 5828, 5832, 5834, 5843, 5844, 5848,

5853, 5854, 5858, 5862, 5866

\c\_colon\_str . . . . . 2587, 3101, 5180

\c\_left\_brace\_str . . . . . 5634, 5641, 5647

\c\_right\_brace\_str . . . . . 5634, 5641, 5647

\str\_case:nn . . . . . 249, 308

\str\_case:nnTF . 1850, 1858, 2415, 2423, 5280, 5289

\str\_clear:N . . . . . 3663, 4683

\str\_count:n . . . . . 236

\str\_if\_empty:NTF . . . . . 1867, 1908, 1935

\str\_if\_eq:nnTF . . . . . 3588, 3634, 5382

\str\_if\_in:nnTF . . . . . 5176

\str\_new:N . . . . . 80, 128, 143, 188

\str\_set:Nn . 675, 681, 687, 706, 707, 708, 2312, 2313,

2345, 2346, 4077, 4080

\str\_use:N . . . . . 3347

\strut . . . . . 3343, 3530

\strutbox . 1354, 1357, 1368, 1369, 1380, 1382, 1397, 1400,

1408, 1409, 1420, 1422, 1437, 1440, 1447, 1448, 1459,

1461, 1476, 1479, 1525, 1528, 1536, 1537, 1545, 1546,

1558, 1560, 1571, 1572, 1575, 1582, 1595, 1603, 1609,

1617, 4137, 4142, 4192, 4200, 4290

## T

## tag commands:

\tag\_mc\_begin:n . . . . . 3977, 4027, 4036

\tag\_mc\_begin\_pop:n . . . . 3993, 4045, 4184, 4186

\tag\_mc\_end: . . . . . 3981, 4031, 4040

\tag\_mc\_end\_push: . . . . . 3974, 4024, 4172

\tag\_resume:n . . 3973, 4023, 4163, 4171, 4240, 4339,

4539, 4603

\tag\_struct\_begin:n . 3975, 3976, 3983, 3984, 3985,

4025, 4026, 4033, 4034, 4035, 4173

\tag\_struct\_end:n 3982, 3989, 3990, 3991, 3992, 4032,

4041, 4042, 4043, 4044, 4183, 4185, 4657, 4919

\tag\_suspend:n . 3994, 4046, 4154, 4165, 4178, 4231,

4331, 4649, 4911

\tag\_tool:n . . . . . 4164

TeX and  $\TeX$  2<sub>ε</sub> commands:

\@auxout . . . . . 423

\@currenvir . . . . . 249, 308

\protected@write . . . . . 423

tex commands:

  \text\_newlinechar:D ..... 2907

text commands:

  \text\_expand:n ..... 5172

\textasteriskcentered ..... 2309, 3367

\textreferencemark ..... 2326

\thepage ..... 429

tl commands:

  \c\_space\_tl 3183, 4266, 5578, 5593, 5616, 5620, 5807, 5808, 5817, 5818, 5878, 5882

  \tl\_clear:N .. 673, 680, 2288, 2362, 2372, 2393, 2401, 2607, 2927, 2928, 3042, 3118, 5022

  \tl\_clear\_new:N ..... 620

  \tl\_const:Nn ..... 46, 604

  \tl\_gclear:N . 356, 357, 358, 1888, 1893, 3017, 3336, 3356, 4575, 4635, 4829

  \tl\_gclear\_new:N ..... 1875

  \tl\_gput\_right:Nn ..... 605

  \tl\_greplace\_all:Nnn ..... 626

  \tl\_gset:Nn 284, 285, 299, 300, 1876, 1889, 1894, 2113, 2931, 3267, 4777

  \tl\_gset\_eq:NN ..... 622, 3263, 4822

  \tl\_if\_blank:nTF 2720, 2738, 2875, 3391, 3409, 3431, 4820, 5446

  \tl\_if\_empty:NTF . 740, 758, 786, 802, 821, 828, 854, 870, 1901, 1906, 1928, 1933, 1991, 2055, 2063, 2092, 2152, 2451, 2482, 2627, 2972, 2994, 3024, 3052, 3128, 3177, 3288, 4346, 5025, 5363

  \tl\_if\_empty:nTF ..... 1956

  \tl\_if\_exist:NTF ..... 1961

  \tl\_if\_novalue:nTF 444, 474, 2734, 3050, 3126, 3162, 3242, 3261, 3269, 3441, 3661, 4109, 4681, 4951, 5023

  \tl\_map\_inline:Nn ..... 227, 623

  \tl\_new:N 38, 39, 40, 43, 48, 49, 52, 53, 59, 61, 62, 64, 65, 101, 102, 103, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 122, 124, 125, 126, 129, 132, 133, 151, 161, 162, 163, 166, 187

  \tl\_put\_left::Ne ..... 2961

  \tl\_put\_left:Nn 2459, 2490, 2612, 2955, 2968, 2974, 2984, 3194, 3234, 4559, 4620, 5044, 5047

  \tl\_put\_right:Nn 621, 750, 794, 810, 862, 2463, 2494, 2541, 2551, 2564, 2579, 2585, 2590, 2614, 2619, 2626, 2629, 2639, 2644, 2647, 2653, 3045, 3048, 3054, 3059, 3086, 3091, 3096, 3099, 3108, 3121, 3124, 3130, 3135, 3145, 5027, 5031

  \tl\_remove\_all:Nn ..... 5362

  \tl\_remove\_once:Nn ..... 2529, 3071

  \tl\_replace\_all:Nnn ..... 625, 5397

  \tl\_reverse:N ..... 2528, 2530, 3070, 3072

  \tl\_set:Nn . 54, 253, 263, 312, 313, 320, 321, 328, 329, 590, 674, 679, 685, 686, 739, 783, 853, 1058, 1072, 1085, 1097, 1990, 2091, 2363, 2373, 2394, 2402, 2684, 2895, 3164, 3209, 3222, 5033, 5056, 5360, 5396, 5466

  \tl\_set\_eq:NN 631, 745, 748, 791, 793, 807, 809, 859, 861, 2527, 3069, 3082, 3453, 3458, 4260, 4262

  \tl\_to\_str:n ..... 1961, 1967, 1972, 5172

  \tl\_trim\_spaces:n ... 621, 5349, 5360, 5366, 5382

  \tl\_use:N 627, 630, 760, 823, 830, 872, 1130, 1134, 1138, 1142, 1146, 1150, 1154, 1158, 1162, 1166, 1170, 1174, 1178, 1182, 1186, 1190, 2517, 2534, 2542, 2553, 2566, 2571, 2582, 3250, 3256, 3284, 3327, 3328, 3335, 3349, 3444, 3448, 3456, 3520, 3521, 3523, 3534, 3825, 3958, 4264, 4270, 4566, 4627, 4833, 4861, 4862, 5104, 5128, 5133, 5227, 5228, 5229, 5230, 5231, 5248, 5345, 5464

token commands:

  \token\_to\_str:N ..... 425

\topsep ..... 3930, 4142

topsep ..... 931

\topskip ..... 1335, 1507

U

\u ..... 230, 2936

\unkern ..... 244

unknown ..... 3377, 3399, 3417

\unskip ..... 243

use commands:

  \use:N ..... 237, 3332, 3353, 3827

  \use:n ..... 1841, 2406, 5178, 5271

  \use\_none:nn ..... 417, 5403

\usecounter ..... 3586, 3631

V

\value .... 1904, 1910, 1917, 1923, 1931, 1937, 1944, 1950

vbox commands:

  \ vbox\_set:Nn ..... 4233

  \ vbox\_set\_top:Nn ..... 4564, 4625

\vspace 991, 1740, 1743, 1754, 1757, 1767, 1769, 1778, 1780, 1789, 1791, 1800, 1802, 1811, 1813, 1822, 1824

W

widest ..... 905

wrap-ans ..... 2290

wrap-key ..... 2290, 2334

wrap-label ..... 644

wrap-label\* ..... 644

wrap-opt ..... 2290, 2334

Z

\z ..... 2936