

The `idxlayout` package*

Thomas Titz†

January 10, 2010

Abstract

The `idxlayout` package offers a key-value interface to configure index layout parameters, e. g. allowing for three-column indexes or for “parent” items and their affiliated subitems being typeset as a single paragraph. The package is responsive to the index-related options and commands of the `KOMA-Script`- and `memoir` classes.

1 Introduction

Every once a while, questions pop up on `comp.text.tex` or other \LaTeX -related newsgroups on how to change properties of the index layout. The most common request is for an index with three instead of two columns. The answer to this – although not necessarily the implementation – is simple: Use a `multicols` environment. An example in the \LaTeX Companion [4, pp. 679–680] shows how to do it.

Besides being cumbersome for unversed \LaTeX users,¹ this approach is prone to break index-related options the document class might provide (e. g. for including the index in the table of contents). To address this, the `idxlayout` package offers a key-value interface to configure index layout parameters while emulating and extending the relevant functionality of the standard, `KOMA-Script`- and `memoir` classes.

Note that `idxlayout` stays on the \LaTeX side of things. If you want to add letter headings to each index group or to change the delimiter between keyword and first page number, you still have to provide a corresponding `MakeIndex` style file.

This manual is typeset according to the conventions of the \LaTeX `DOCSTRIP` utility which enables the automatic extraction of the \LaTeX macro source files [4, ch. 14]. Section 2 describes the usage of the package, while the optional section 3 contains the commented source code.

*This document corresponds to `idxlayout` v0.4, dated 2010/01/10.

†E-mail: `thomas.titz@chello.at`

¹In the words of user `InsideMan` on `mrunix.de`, November 27, 2009 (authors translation): “I didn’t believe this to be so laborious. I’d rather have guessed there is a package option to cater for it.”

Index	Index	Index
Dion, 2, 3, 5, 8, 13, 21, 1957–1960 and the Belmonts, 1957–1960	Dion, 2, 3, 5, 8, 13, 21, 1957–1960 and the Belmonts, 1957–1960	Dion, 2, 3, 5, 8, 13, 21, 1957–1960; and the Belmonts, 1957–1960

Figure 1: Option `itemlayout` with values `abshang`, `relhang` and `singlepar`

2 Using the `idxlayout` package

2.1 Default behaviour

If you just add `\usepackage{idxlayout}` to your document preamble, the package will keep up the index-related behaviour of the document class while correcting bugs and taking advantage of `multicol` features. This amounts to the following changes:

- Columns will be balanced on the last index page.
- If you are using a class that lacks the `\chapter` command (or `memoir` with the option `article`), the index will behave like a normal `\section*`, i. e. it will not automatically start a new page.² The first index page will also not use a special pagestyle like `plain` appropriate only for chapter headings.
- When on top of the page, the vertical positioning of the index heading will not differ from that of other headings typeset in `onecolumn` mode, i. e. \LaTeX bug 3126 (`\twocolumn` interfering with `\topskip`) will be avoided.
- With the standard classes `article`, `book` and `report`, the space between index columns will be 35 pt as intended.³

2.2 Options and commands

The options provided by `idxlayout` use the key–value interface ‘ $\langle key \rangle = \langle value \rangle$ ’. Unless indicated otherwise, any valid \TeX dimension is a permitted value.

columns The number of columns used for typesetting the index. The permitted values range from 1 to 10, the default value is 2. A single-column index is e. g. suitable “for an index of first lines in a book of poetry” [5, p. XXIX].

columnsep The space between index columns. The default value is 35 pt. As this is required for compatibility rather than suggested by typographic considerations, you may want to reduce the value of `columnsep` to, say, 1 to 1.5 times the value of `indentunit`.

²However, if you typeset your document in `twocolumn` mode, `idxlayout` will temporarily switch to `onecolumn`, causing a page break in the process.

³In the current version (v1.4h) of these classes, the length `\columnsep` is set to 35 pt, but only after `\twocolumn` has been triggered. The KOMA-Script- and `memoir` classes have either corrected or never introduced this bug. See [1] for details.

Table 1: Default values of options `hangindent`, `subindent` and `subsubindent`

Value of option <code>itemlayout</code>	Default value (multiple of <code>indentunit</code>)		
	<code>hangindent</code>	<code>subindent</code>	<code>subsubindent</code>
<code>abshang</code>	2	1	1.5
<code>relhang</code>	1.5	1	2
<code>singlepar</code>	1	–	–

rule The thickness of a vertical rule between index columns. The default value is 0pt, i. e. no rule.

itemlayout The layout of index items, subitems and subsubitems. The permitted values, which are described below, are `abshang`, `relhang` and `singlepar`. The default value is `abshang` (which corresponds to standard L^AT_EX behaviour). See figure 1 on the preceding page for examples of the various layouts.

abshang Index items feature a hanging indentation of value `hangindent`. Subitems and subsubitems resemble items except that their first line is indented by the values of `subindent` and `subsubindent`, respectively.

relhang Differing from `abshang`, the hanging indentations of subitems and subsubitems equal the sum of `hangindent` and the respective first-line indentations.

singlepar “Parent” items and their affiliated subitems (and, if present, subsubitems) are typeset as a single paragraph, with a semicolon plus an interword space being used as the default delimiter.⁴

indentunit A length that serves as a building block for other indentation lengths. The default value is 20pt. The default values of the options `hangindent`, `subindent` and `subsubindent` will be set in accordance with the chosen values of `itemlayout` and `indentunit` (see table 1).

hangindent Use this option to override the default value of `hangindent`.

subindent Use this option to override the default value of `subindent`.

subsubindent Use this option to override the default value of `subsubindent`.

initsep The vertical space inserted between index groups with differing initial letters. The default value is 10pt plus 5pt minus 3pt.

notesep The vertical space inserted between a possible prenote and the first index entry. The default value is one times the value of `initsep`.

⁴This blends in well with the default behaviour of `MakelIndex` to put a comma plus an interword space between keyword and first page number.

minspace The vertical space needed on the current page to prevent the index from starting a new page. The default value is 80pt. You may want to increase the value of **minspace** if the index includes a prenote containing several lines.

font The font used for typesetting the index. The permitted values are **current** (i. e. the font in effect at the time of `\printindex`), **normalsize**, **small** and **footnotesize**. The default value is **current**.

justific The justification of index columns. The permitted values are **standard** (i. e. left and right margin justified), **raggedright** and **RaggedRight** (which uses the command of the same name of the **ragged2e** package). The default value is **standard**.

totoc A boolean option that controls whether the index is included in the table of contents. The default value is **false**. You may simply specify **totoc**, which is equivalent to **totoc=true**.

`\setindexprenote` Use the command `\setindexprenote{<Some text>}` to typeset explanatory text after the index heading. As a default, the prenote (which may span several paragraphs) is set in the same font as the index. Font changes within the argument of `\setindexprenote` won't affect the index proper. A vertical space controlled by the **notesep** option is inserted between the prenote and the first index entry. With multiple indexes, use the command `\noindexprenote` to reset the prenote text to empty and to take care of the proper spacing between index heading and first entry.

`\idxlayout` The command `\idxlayout`, which takes a comma-separated list of key–value options as argument, allows to change index layout parameters within the document. A typographically sound example is provided by [4] which typesets a general index in two columns and then switches to three columns for an index of names.

`\indexfont` If the permitted values of the options **font** and **justific** do not cover your needs, you can resort to redefining the commands `\indexfont` and `\indexjustific`, e. g. by including `\renewcommand*{\indexfont}{\tiny\bfseries}` in your document preamble.⁵

`\indexsubsdelim` Redefine the command `\indexsubsdelim` if you happen to dislike the default delimiter used with the option `itemlayout=singlepar`.

2.3 Compatibility issues and notes

`idxlayout` has been tested with the standard classes **article**, **book** and **report** (v1.4h), the KOMA-Script-classes **scrartcl**, **scrbook** and **scrreprt** (v3.04a) and the **memoir** class (v1.61803398c). It is responsive to their index-related options and commands:

- The KOMA-Script options for including the index in the table of contents (e. g. `index=totoc`) and the respective **memoir** commands (e. g. `\indexintoc`) are recognized and can be mixed with the `idxlayout` option **totoc**.

⁵See the file `idxlayout.dtx` for the redefinition of `\indexjustific` carried out in order to use `idxlayout` with the `ltxdoc` class.

- The memoir lengths `\indexcolsep` and `\indexrule` are supported, and it is possible to alternate between setting these lengths and using the respective `idxlayout` options `columnsep` and `rule`.
- The memoir commands `\onecolindex` and `\twocolindex` are recognized. (For an index with three or more columns, use the `idxlayout` option `columns`.)
- The KOMA-Script command `\setindexpreamble` and the memoir command `\preindexhook` are supported. Note that both of them operate independently of the `idxlayout` commands `\setindexprenote` and `\noindexprenote`.

When using `idxlayout` together with the `splitidx` package, issue the `splitidx` command `\extendtheindex` only after having loaded `idxlayout`.

If you use the `tocbibind` package, you must load it before `idxlayout`. The influence of `tocbibind` on including the index in the table of contents will be removed, so use the `idxlayout` option `totoc` instead.⁶

The packages `multind` (from 1991) and `index` (from 1995/2004) are not supported. If you’re feeling so lucky as to use `index`, load it before `idxlayout`.

The `ragged2e` package must be loaded before `idxlayout` in order to use the option `justific=RaggedRight`.

As `idxlayout` is based on the `multicol` package, the appearance of the index is affected by any `multicol` parameter not overridden by `idxlayout`. Use the command `\raggedcolumns` if you aim at typesetting the index on a grid.

3 The package code

3.1 Preliminaries

Announce the name and version of the package, which requires \LaTeX 2_ϵ .

```
1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{idxlayout}[2010/01/10 v0.4 Configurable index layout]
```

Besides `multicol`, we require the packages `kvoptions` (to conveniently specify options as key–value pairs) and `etoolbox` (e.g. for length assignments and boolean switch handling). We also need $\epsilon\text{-TeX}$, whose presence is tested by `etoolbox`.

```
3 \RequirePackage{etoolbox,kvoptions,multicol}
```

We use `ila@` as a short prefix to internal macros defined by `kvoptions`.

```
4 \SetupKeyvalOptions{family=ila,prefix=ila@}
```

`\ila@defradiokey` `kvoptions` does not provide “radio buttons” – options that take a list of three or more values with associated functions and yield an error message if an unknown value is given. However, in a series of messages on `comp.text.tex` [2], Heiko Oberdiek (author of `kvoptions`) and Joseph Wright showed how radio button keys can be created using `\define@key` (from the `keyval` package, which is loaded by `kvoptions`)

⁶Just in case you ask, numbering the index heading is not supported by `idxlayout` because it is a bad idea from a typographic point of view. See [3, pp. 86–87] for details.

and the ε -TeX primitive `\ifcsname`. What follows is basically a shameless rip-off of Heikos and Josephs good advice.

The command `\ila@defradiokey` takes one optional and two mandatory arguments. The first mandatory argument is the key to be defined, the second is a “short key” which must be used as a second prefix after `ila@` when defining the functions associated with each value. The optional argument specifies a default value “which is then used if the user does not supply one” [6, p. 1002].

```

5 \newcommand*{\ila@defradiokey}[3] [] {%
6   \define@key{ila}{#2}[#1]{%
7     \ifcsname ila@#3@##1\endcsname
8       \csname ila@#3@##1\expandafter\endcsname
9     \else
10      \PackageError{idxlayout}{Unknown value ##1 for option #2}%
11    \fi
12  }%
13 }
```

3.2 Options and commands

`\theidxcols` We define the key `columns` to control the number of index columns and initially set it to 2. (Actually, we use the counter `idxcols` to store the number so that anything but an integer will yield a L^AT_EX error message.) If the `memoir` class is used, we check the state of its boolean switch `onecolindex` and set `columns` accordingly. We also append code to `\onecolindex` and `\twocolindex` so that these `memoir` commands will set `columns` to the respective number.

```

14 \newcounter{idxcols}
15 \define@key{ila}{columns}{\setcounter{idxcols}{#1}}
16 \setkeys{ila}{columns=2}
17 \@ifclassloaded{memoir}{%
18   \ifbool{onecolindex}{%
19     \setkeys{ila}{columns=1}%
20   }{%
21   }%
22   \appto{\onecolindex}{\setkeys{ila}{columns=1}}%
23   \appto{\twocolindex}{\setkeys{ila}{columns=2}}%
```

`\indexcolsep` If any other class than `memoir` is used, we define the lengths `\indexcolsep` and `\indexrule` and set them to appropriate values. The keys `columnsep` and `rule` will be responsible for changing these lengths. We don’t set these keys here in order not to override any changes carried out before loading `idxlayout`.

```

24 }{%
25   \newlength{\indexcolsep}%
26   \setlength{\indexcolsep}{35\p}%
27   \newlength{\indexrule}%
28   \setlength{\indexrule}{\z@}%
29 }
30 \define@key{ila}{columnsep}{\setlength{\indexcolsep}{#1}}
31 \define@key{ila}{rule}{\setlength{\indexrule}{#1}}
```

`\ila@indentunit` The key `indentunit` is also defined and set using `keyval` commands so that its associated length can be used to set other lengths.

```

32 \newlength{\ila@indentunit}
33 \define@key{ila}{indentunit}{\setlength{\ila@indentunit}{#1}}
34 \setkeys{ila}{indentunit=20\p@}

```

`\ila@hangindent` We finally use `kvoptions` to define the keys `hangindent`, `subindent` and `subsubindent`.
`\ila@subindent` Setting their default values is deferred to the definition of the functions of the
`\ila@subsubindent` “radio button” key `itemlayout`.

```

35 \DeclareStringOption{hangindent}
36 \DeclareStringOption{subindent}
37 \DeclareStringOption{subsubindent}

```

`\@idxitem` `\@idxitem` is only slightly changed: Its hanging indentation is controlled by the key `hangindent` instead of being a fixed value.

```

38 \renewcommand{\@idxitem}{\par\setlength{\hangindent}{\ila@hangindent}}

```

`\subitem` The redefinitions of `\subitem` and `\subsubitem` have to be done separately for
`\subsubitem` each permitted value of `itemlayout`. Beforehand, we have to set the values of `hangindent`, `subindent` and `subsubindent` to appropriate multiples of `indentunit`.

```

39 \def\ila@it@abshang{%
40   \renewcommand*{\ila@hangindent}{2\ila@indentunit}%
41   \renewcommand*{\ila@subindent}{\ila@indentunit}%
42   \renewcommand*{\ila@subsubindent}{1.5\ila@indentunit}%
43   \renewcommand{\subitem}{\@idxitem\hspace*{\ila@subindent}}%
44   \renewcommand{\subsubitem}{\@idxitem\hspace*{\ila@subsubindent}}%
45 }

```

Defining the function of `relhang` also starts with setting various indentation values. `\deflength` from the `etoolbox` package comes in handy for the calculation of the hanging indentations of `\subitem` and `\subsubitem`.

```

46 \def\ila@it@relhang{%
47   \renewcommand*{\ila@hangindent}{1.5\ila@indentunit}%
48   \renewcommand*{\ila@subindent}{\ila@indentunit}%
49   \renewcommand*{\ila@subsubindent}{2\ila@indentunit}%
50   \renewcommand{\subitem}{%
51     \par
52     \deflength{\hangindent}{\ila@hangindent + \ila@subindent}%
53     \hspace*{\ila@subindent}%
54   }%
55   \renewcommand{\subsubitem}{%
56     \par
57     \deflength{\hangindent}{\ila@hangindent + \ila@subsubindent}%
58     \hspace*{\ila@subsubindent}%
59   }%
60 }

```

`\indexsubsdelim` For `singlepar`, only the value of `hangindent` needs to be set. We define `\subitem` and `\subsubitem` as `\unskip` (to remove the space generated by the preceding linebreak in the ind-file) plus `\indexsubsdelim` which stores the delimiter.

```

61 \newcommand*{\indexsubsdelim}{; }
62 \def\ila@it@singlepar{%
63   \renewcommand*{\ila@hangindent}{\ila@indentunit}%
64   \renewcommand{\subitem}{\unskip\indexsubsdelim}%
65   \renewcommand{\subsubitem}{\unskip\indexsubsdelim}%
66 }

```

At last, we define the “radio button” key `itemlayout`, which defaults to `abshang`.

```

67 \ila@defradiokey{itemlayout}{it}
68 \setkeys{ila}{itemlayout=abshang}

```

`\ila@initsep` We use `keyval` commands to define and set the key `initsep`, then `kvoptions` to define

`\ila@notesep` the keys `notesep` and `minspace`, this time also specifying default values.

`\ila@minspace`

```

69 \newlength{\ila@initsep}
70 \define@key{ila}{initsep}{\setlength{\ila@initsep}{#1}}
71 \setkeys{ila}{initsep=10\p@ \@plus 5\p@ \@minus 3\p@}
72 \DeclareStringOption[\ila@initsep]{notesep}
73 \DeclareStringOption[80\p@]{minspace}

```

`\indexspace` The fixed vertical space inserted by `\indexspace` is replaced by the value of `initsep`.

```

74 \renewcommand{\indexspace}{\par\vspace{\ila@initsep}}

```

`\indexfont` The “radio button” key font produces an appropriate definition of `\indexfont`.

```

75 \newcommand*{\indexfont}{}
76 \def\ila@fo@current{\renewcommand*{\indexfont}{}}
77 \def\ila@fo@normalsize{\renewcommand*{\indexfont}{\normalsize}}
78 \def\ila@fo@small{\renewcommand*{\indexfont}{\small}}
79 \def\ila@fo@footnotesize{\renewcommand*{\indexfont}{\footnotesize}}
80 \ila@defradiokey{font}{fo}
81 \setkeys{ila}{font=current}

```

`\indexjustific` The “radio button” key `justific` is responsible for defining `\indexjustific`. With the value `standard` selected, we (like the KOMA-Script-classes) also explicitly set `\parfillskip` to its default value of 0pt plus 1fil. This way, resettings done outside the `theindex` environment won’t affect the justification of index columns. If `RaggedRight` is chosen without the command of the same name being defined, we issue a warning message and use `\raggedright` instead.

```

82 \newcommand*{\indexjustific}{}
83 \def\ila@ju@standard{%
84   \renewcommand*{\indexjustific}{%
85     \setlength{\parindent}{\z@}%
86     \setlength{\parfillskip}{\z@ \@plus 1fil}%
87   }%
88 }
89 \def\ila@ju@raggedright{\renewcommand*{\indexjustific}{\raggedright}}

```



```

90 \def\ila@ju@RaggedRight{%
91   \ifundef{\RaggedRight}{%
92     \PackageWarning{idxlayout}{%
93       Command RaggedRight not defined!\MessageBreak
94       I'm using raggedright instead}%
95   }%
96   \renewcommand*{\indexjustific}{\raggedright}%
97 }{%
98   \renewcommand*{\indexjustific}{\RaggedRight}%
99 }
100 \ila@defradiokey{justific}{ju}
101 \setkeys{ila}{justific=standard}

```

`\ila@prenote` We use `\setindexprenote` to specify the text of a possible prenote. `\ila@prenote` typesets the text within a group (to keep font changes local) and inserts the value of `notesep` as vertical space. `\noindexprenote` defines `\ila@prenote` as `\relax` (which also means that the control sequence test `\ifundef` will yield true).

```

102 \newcommand{\ila@prenote}{}
103 \newcommand{\setindexprenote}[1]{%
104   \def\ila@prenote{%
105     \begingroup#1\par\nobreak\endgroup
106     \vspace{\ila@notesep}%
107   }%
108 }
109 \newcommand*{\noindexprenote}{\let\ila@prenote\relax}
110 \noindexprenote

```

3.3 Emulating specific document classes

`\ila@classtype` We store the type of document class we use, differentiating between the KOMA-Script-classes (which define `\KOMAClassName`), the memoir class and all other (notably the standard) classes.

```

111 \newcommand*{\ila@classtype}{0}
112 \@ifclassloaded{memoir}{%
113   \def\ila@classtype{2}%
114 }{%
115   \ifundef{\KOMAClassName}{%
116     }{%
117       \def\ila@classtype{1}%
118     }%
119 }

```

For any but the KOMA-Script- and memoir classes, we define the boolean option `totoc` which is responsible for including the index in the table of contents. If not set, this option defaults to false.

```

120 \ifcase\ila@classtype\relax
121   \DeclareBoolOption{totoc}%

```

`\ila@prologue` The command `\ila@prologue` comprises actions at the beginning of the `theindex` environment that must be tailored to the type of document class. For “other” classes, the code resembles that of the standard classes `article`, `book` and `report`.

Unlike `\twocolumn`, `\begin{multicols}` does not automatically insert a page-break, so we use `\chapter*` instead of `\@makeschapterhead`. With that, we also eliminate the need to set `\thispagestyle{plain}` for classes containing the `\chapter` command. (For other classes, we don’t want the first index page to show a special pagestyle.) Furthermore, we take our boolean option `totoc` into account. Note that with version 6.79z of the `hyperref` package, we can do without inserting `\phantomsection` before `\addcontentsline`.

```

122 \def\ila@prologue{%
123   \ifundef{\chapter}{%
124     \section*{\indexname}%
125     \ifbool{ila@totoc}{%
126       \addcontentsline{toc}{section}{\indexname}%
127     }{%
128     }%
129   }{%
130     \chapter*{\indexname}%
131     \ifbool{ila@totoc}{%
132       \addcontentsline{toc}{chapter}{\indexname}%
133     }{%
134     }%
135   }%
136   \@mkboth{\MakeUppercase\indexname}{\MakeUppercase\indexname}%
137 }

```

For the KOMA-Script-classes `scrartcl`, `scrbook` and `scrreprt`, we create `totoc` as a pseudo boolean key using `\ila@defradiokey`. If not set, this key will respect the default behaviour of the class plus any changes carried out before loading `idxlayout`.

```

138 \or
139 \def\ila@tc@true{\KOMAoptions{index=totoc}}%
140 \def\ila@tc@false{\KOMAoptions{index=nottotoc}}%
141 \ila@defradiokey[true]{totoc}{tc}%

```

Again, `\ila@prologue` is very much a recoding exercise. Note that the KOMA-script-command `\idx@heading` is responsive to `\KOMAoptions` and therefore to our key `totoc`. For `scrbook` and `scrreprt`, we only use `\setchapterpreamble` if `\index@preamble` is not defined as `\relax`. Thus, we avoid excessive vertical space between the index heading and the first entry. For `scrartcl`, we drop the special pagestyle of the first index page.

```

142 \def\ila@prologue{%
143   \ifundef{\chapter}{%
144   }{%
145     \ifundef{\index@preamble}{%
146     }{%
147       \setchapterpreamble{\index@preamble}%
148     }%

```

```

149     }%
150     \idx@@heading{\indexname}%
151     \@mkboth{\indexname}{\indexname}%
152     \ifundef{\chapter}{%
153         \index@preamble
154     }{%
155         \thispagestyle{\indexpagestyle}%
156     }%
157 }

```

For the memoir class, our pseudo boolean key `totoc` will also respect prior settings.

```

158 \or
159 \def\ila@tc@true{\boolfalse{noindexintoc}}%
160 \def\ila@tc@false{\booltrue{noindexintoc}}%
161 \ila@defradiokey[true]{totoc}{tc}%

```

Another recoding exercise. Note that with `memoir` and its `article` option, `\chapter` resembles `\section`. Again, we get rid of special pagestyles, if necessary, and use neither `\@makeschapterhead` nor `\phantomsection`.

```

162 \def\ila@prologue{%
163     \chapter*{\indexname}%
164     \ifbool{noindexintoc}{%
165         }{%
166             \addcontentsline{toc}{chapter}{\indexname}%
167         }%
168     \ifbool{artopt}{%
169         }{%
170             \thispagestyle{indextitlepagestyle}%
171         }%
172     \indexmark
173     \preindexhook
174 }
175 \fi

```

Having defined the option `totoc`, we are able to process our options.

```

176 \ProcessKeyvalOptions*

```

`\idxlayout` We define `\idxlayout` as a shorthand for `\setkeys{ila}`.

```

177 \newcommand*{\idxlayout}[1]{\setkeys{ila}{#1}}

```

3.4 Redefining the environment `theindex`

`theindex` At the beginning of `theindex`, we check if we are in `twocolumn` mode, store the result in `@restonecol` and switch to `onecolumn` mode, if necessary.

```

178 \renewenvironment{theindex}{%
179     \ifbool{@twocolumn}{%
180         \boolfalse{@restonecol}%
181         \onecolumn
182     }{%
183         \booltrue{@restonecol}%
184     }%

```

`\multicolsep` is set to 0pt, again in order to avoid excessive vertical space after the index heading. Also, the memoir lengths `\indexcolsep` and `\indexrule` (now responsive to the `idxlayout` options `columnsep` and `rule`) are put in charge.

```
185 \setlength{\multicolsep}{\z@}%
186 \setlength{\columnsep}{\indexcolsep}%
187 \setlength{\columnseprule}{\indexrule}%
```

If a single-column index is requested, we simply execute `\ila@prologue`, switch to `\indexfont` and typeset a possible prenote. In case of two or more columns, we do all this inside the first optional argument of a properly chosen `multicols` environment and specify the value of `minspace` as its second optional argument.

```
188 \ifnumcomp{\theidxcols}{<}{\tw@}{%
189   \ila@prologue\indexfont\ila@prenote
190 }{%
191   \begin{multicols}{\theidxcols}%
192   [\ila@prologue\indexfont\ila@prenote][\ila@minspace]%
193 }%
```

We set `\parskip` to 0pt plus 0.3pt (exactly as in the standard classes), switch to the proper justification and – last, but not least – redefine `\item`.

```
194 \setlength{\parskip}{\z@ \@plus 0.3\p@}%
195 \indexjustific
196 \let\item\@idxitem
```

At the end of `theindex`, we close a possible `multicols` environment and switch back to `twocolumn` mode, if necessary.

```
197 }{%
198 \ifnumcomp{\theidxcols}{<}{\tw@}{%
199 }{%
200   \end{multicols}%
201 }%
202 \ifbool{@restonecol}{%
203 }{%
204   \twocolumn
205 }%
206 }
```

The end of the package.

```
207 \endinput
```

References

- [1] “columnsep in theindex (TLC2 and memoir)”. Series of messages on `comp.text.tex`, May 22–29, 2005. http://groups.google.com/group/comp.text.tex/browse_thread/thread/937744003315b784 (accessed January 2, 2010).
- [2] “\DeclareStringOption”. Series of messages on `comp.text.tex`, December 14, 2009. http://groups.google.com/group/comp.text.tex/browse_thread/thread/1fb53938ac0026c5 (accessed January 2, 2010).

- [3] Kohm, Markus and Jens-Uwe Morawski. *KOMA-Script: Eine Sammlung von Klassen und Paketen für L^AT_EX 2_ε*. 3rd edition. Berlin: lehmanns media, 2008.
- [4] Mittelbach, Frank and Michel Goossens. *The L^AT_EX Companion*. 2nd edition. Boston: Addison-Wesley, 2004.
- [5] Wilson, Peter. *The memoir class for configurable typesetting: User guide*. 8th edition. Published by the author, 2009. <http://www.ctan.org/tex-archive/macros/latex/contrib/memoir/memman.pdf> (accessed January 2, 2010).
- [6] Wright, Joseph and Christian Feuersänger. “Implementing key–value input: An introduction”. Preliminary draft, March 17, 2009. <http://www.texdev.net/wp-content/uploads/2009/03/keyval.pdf> (accessed January 2, 2010).

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

Symbols	E	<code>\ila@hangindent</code> ..
<code>\@idxitem</code> <u>38</u> , 43, 44, 196	environments: <u>35</u> , 38,
<code>\@ifclassloaded</code> 17, 112	<code>theindex</code> <u>178</u>	40, 47, 52, 57, 63
<code>\@mkboth</code> 136, 151		<code>\ila@indentunit</code> <u>32</u> ,
	F	40–42, 47–49, 63
A	font (option) 4	<code>\ila@initsep</code> ... <u>69</u> , 74
<code>\addcontentsline</code> .	<code>\footnotesize</code> 79	<code>\ila@it@abshang</code> ... 39
.... 126, 132, 166		<code>\ila@it@relhang</code> ... 46
<code>\appto</code> 22, 23	H	<code>\ila@it@singlepar</code> . 62
	<code>\hangindent</code> . 38, 52, 57	<code>\ila@ju@RaggedRight</code> 90
B	hangindent (option) .. 3	<code>\ila@ju@raggedright</code> 89
<code>\boolfalse</code> .. 159, 180		<code>\ila@ju@standard</code> .. 83
<code>\booltrue</code> ... 160, 183	I	<code>\ila@minspace</code> . <u>69</u> , 192
	<code>\idx@@heading</code> 150	<code>\ila@notesep</code> .. <u>69</u> , 106
C	<code>\idxlayout</code> 4, <u>177</u>	<code>\ila@prenote</code>
<code>\chapter</code> 123,	<code>\ifbool</code> 18, 125, 131, <u>102</u> , 189, 192
130, 143, 152, 163	164, 168, 179, 202	<code>\ila@prologue</code> . <u>122</u> ,
columns (option) 2	<code>\ifnumcomp</code> .. 188, 198	142, 162, 189, 192
<code>\columnsep</code> 186	<code>\ifundef</code> .. 91, 115,	<code>\ila@subindent</code> . <u>35</u> ,
columnsep (option) .. 2	123, 143, 145, 152	41, 43, 48, 52, 53
<code>\columnseprule</code> ... 187	<code>\ila@classtype</code> <u>111</u> , 120	<code>\ila@subsubindent</code>
	<code>\ila@defradiokey</code> <u>35</u> ,
D 5, 67,	42, 44, 49, 57, 58
<code>\DeclareBoolOption</code> 121	80, 100, 141, 161	<code>\ila@tc@false</code> 140, 160
<code>\DeclareStringOption</code>	<code>\ila@fo@current</code> ... 76	<code>\ila@tc@true</code> . 139, 159
.... 35–37, 72, 73	<code>\ila@fo@footnotesize</code>	indentunit (option) .. 3
<code>\define@key</code> 6, 79	<code>\index@preamble</code> ..
15, 30, 31, 33, 70	<code>\ila@fo@normalsize</code> . 77 145, 147, 153
<code>\deflength</code> 52, 57	<code>\ila@fo@small</code> 78	<code>\indexcolsep</code> .. <u>24</u> , 186

<code>\indexfont</code>	4, <u>75</u> , 189, 192				
<code>\indexjustific</code>	4, <u>82</u> , 195				
<code>\indexmark</code> 172				
<code>\indexname</code>	124, 126, 130, 132, 136, 150, 151, 163, 166				
<code>\indexpagestyle</code>	.. 155				
<code>\indexrule</code>	... <u>24</u> , 187				
<code>\indexspace</code> <u>74</u>				
<code>\indexsubsdelim</code>	. 4, <u>61</u>				
<code>initsep (option)</code> 3				
<code>itemlayout (option)</code>	.. 3				
J					
<code>justific (option)</code> 4				
K					
<code>\KOMAClassName</code>	... 115				
<code>\KOMAoptions</code>	. 139, 140				
M					
<code>\MakeUppercase</code>	... 136				
<code>minspace (option)</code>	... 4				
<code>\multicolsep</code> 185				
N					
<code>\noindexprenote</code>	4, <u>102</u>				
<code>\normalsize</code> 77				
<code>notesep (option)</code> 3				
O					
<code>\onecolindex</code> <u>14</u>				
<code>\onecolumn</code> 181				
options:					
columns 2				
columnsep 2				
font 4				
hangindent 3				
indentunit 3				
initsep 3				
itemlayout 3				
justific 4				
minspace 4				
notesep 3				
rule 3				
subindent 3				
subsubindent 3				
totoc 4				
P					
<code>\PackageError</code> 10				
<code>\PackageWarning</code>	... 92				
<code>\parfillskip</code> 86				
<code>\parindent</code> 85				
<code>\parskip</code> 194				
<code>\preindexhook</code> 173				
<code>\ProcessKeyvalOptions</code> 176				
R					
<code>\RaggedRight</code>	... 91, 98				
<code>\raggedright</code>	... 89, 96				
<code>rule (option)</code> 3				
S					
<code>\section</code> 124				
<code>\setchapterpreamble</code>	147				
<code>\setindexprenote</code>	4, <u>102</u>				
<code>\setkeys</code> 16, 19, 22, 23, 34, 68, 71, 81, 101, 177				
<code>\SetupKeyvalOptions</code>	. 4				
<code>\small</code> 78				
<code>subindent (option)</code>	... 3				
<code>\subitem</code>	... <u>39</u> , 50, 64				
<code>subsubindent (option)</code>	3				
<code>\subsubitem</code>	. <u>39</u> , 55, 65				
T					
<code>\theidxcols</code> <u>14</u> , 188, 191, 198				
<code>theindex</code>	(environ- ment) <u>178</u>				
<code>\thispagestyle</code>	155, 170				
<code>totoc (option)</code> 4				
<code>\twocolindex</code> <u>14</u>				
<code>\twocolumn</code> 204				
U					
<code>\unskip</code> 64, 65				

Change History

v0.4

General: Initial public release 1