

The `idxlayout` package*

Thomas Titz[†]

March 30, 2012

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 $\text{\LaTeX}$ -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 $\text{\LaTeX}$ Companion [4, pp. 679–680] shows how to do it.

Besides being cumbersome for unversed $\text{\LaTeX}$ users,¹ the do-it-yourself 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 $\text{\LaTeX}$ 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 document corresponds to `idxlayout` v0.4d, dated 2012/03/30.

[†]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 basically 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. L^AT_EX 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 ‘`<key>=<value>`’. Set them with `\usepackage[<options>]{idxlayout}`. For boolean options, the default value is `false`, and specifying `<key>` is equivalent to `<key>=true`.

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].

unbalanced A boolean option that, if enabled, switches off the balancing of columns on the last index page.

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	–	–

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`.

rule The thickness of a vertical rule between index columns. The default value is 0 pt, 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 20 pt. You may want to match the value of `indentunit` with that of the T_EX length `\parindent`. 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`.

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

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

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

columnnote A boolean option that controls whether a possible prenote is typeset as part of the first index column.

minspace The vertical space needed on the current page to prevent the index from starting a new page. The default value is 80 pt. 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 correspondent command of the automatically loaded **ragged2e** package).⁵ The default value is **standard**.

totoc A boolean option that controls whether the index is included in the table of contents. The default value of this option is set according to the standard behaviour of the document class (**true** for **memoir**, **false** for other classes).

`\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 **noteseq** 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.⁶

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

`\indexstheadcase` In the standard classes, the index header is typeset using `\MakeUppercase`. `idxlayout` allows you to change this by redefining `\indexstheadcase`.

⁵Before `idxlayout v0.4c`, `justific=RaggedRight` erroneously utilized `\raggedright`.

⁶See the file `idxlayout.dtx` for the redefinition of `\indexjustific` carried out in order to use `idxlayout` with the `ltxdoc` class.

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` (v3.10a) and the `memoir` class (v3.6j). 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`.
- 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.⁷

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.⁸

Another package which must not be loaded after `idxlayout` is `imakeidx` which allows to create indexes during the main typesetting run.

The `index` package also has to be loaded before `idxlayout`. You may use the newly-defined second optional argument of `\printindex` to override the prenote text of `\setindexprenote` for individual indexes.

The `multind` package (from 1991) is not supported.

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 L^AT_EX 2_ε.

```
1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{idxlayout}[2012/03/30 v0.4d Configurable index layout]
```

⁷Because these commands do not automatically insert vertical space between their argument and the first index entry, they have been made to operate independently of the `idxlayout` commands `\setindexprenote` and `\noindexprenote`. With `idxlayout` v0.4b and the `scrartcl` class, `\setindexpreamble` will correctly typeset the preamble as part of the first index column.

⁸Just 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.

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 ε - $\text{T}_{\text{E}}\text{X}$, 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`) and the ε - $\text{T}_{\text{E}}\text{X}$ 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. 111].

```
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 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 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` `\indexrule` 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}}

```

`kvoptions` is used to define the boolean option `unbalanced` which is responsible for switching off the multicolumn balancing output routine.

```

32 \DeclareBoolOption{unbalanced}

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

40 \def\ila@it@abshang{%
41   \renewcommand*{\ila@hangindent}{2\ila@indentunit}%
42   \renewcommand*{\ila@subindent}{\ila@indentunit}%
43   \renewcommand*{\ila@subsubindent}{1.5\ila@indentunit}%
44   \renewcommand{\subitem}{\@idxitem\hspace*{\ila@subindent}}%
45   \renewcommand{\subsubitem}{\@idxitem\hspace*{\ila@subsubindent}}%
46 }

```

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`.

```

47 \def\ila@it@relhang{%
48   \renewcommand*{\ila@hangindent}{1.5\ila@indentunit}%
49   \renewcommand*{\ila@subindent}{\ila@indentunit}%
50   \renewcommand*{\ila@subsubindent}{2\ila@indentunit}%
51   \renewcommand{\subitem}{%
52     \par
53     \deflength{\hangindent}{\ila@hangindent + \ila@subindent}%
54     \hspace*{\ila@subindent}%
55   }%
56   \renewcommand{\subsubitem}{%
57     \par
58     \deflength{\hangindent}{\ila@hangindent + \ila@subsubindent}%
59     \hspace*{\ila@subsubindent}%
60   }%
61 }

```

`\indexsubselim` 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 `\indexsubselim` which stores the delimiter.

```

62 \newcommand*{\indexsubselim}{; }
63 \def\ila@it@singlepar{%
64   \renewcommand*{\ila@hangindent}{\ila@indentunit}%
65   \renewcommand{\subitem}{\unskip\indexsubselim}%
66   \renewcommand{\subsubitem}{\unskip\indexsubselim}%
67 }

```

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

```

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

```

`\ila@initsep` We use `keyval` commands to define and set the key `initsep`, then `kvoptions` to define the keys `notesep` and `minspace`, this time also specifying default values.

```

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

```

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

```

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

```

The boolean option `columnnote` controls whether a possible prenote is placed within the first index column.

```

76 \DeclareBoolOption{columnnote}

```

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

```

77 \newcommand*{\indexfont}{}
78 \def\ila@fo@current{\renewcommand*{\indexfont}{}}
79 \def\ila@fo@normalsize{\renewcommand*{\indexfont}{\normalsize}}

```

```

80 \def\ila@fo@small{\renewcommand*{\indexfont}{\small}}
81 \def\ila@fo@footnotesize{\renewcommand*{\indexfont}{\footnotesize}}
82 \ila@defradiokey{font}{fo}
83 \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. We try to load the `ragged2e` package at the end of the preamble in order to prevent option clashes. If the package is not available, we issue a warning message and use `\raggedright` instead of `\RaggedRight`.

```

84 \newcommand*{\indexjustific}{}
85 \def\ila@ju@standard{%
86   \renewcommand*{\indexjustific}{%
87     \setlength{\parindent}{\z@}%
88     \setlength{\parfillskip}{\z@ \@plus 1fil}%
89   }%
90 }
91 \def\ila@ju@raggedright{\renewcommand*{\indexjustific}{\raggedright}}
92 \newcommand*{\ila@RaggedRight}{}
93 \def\ila@ju@RaggedRight{%
94   \renewcommand*{\indexjustific}{\ila@RaggedRight}%
95 }
96 \AtEndPreamble{%
97   \IfFileExists{ragged2e.sty}{%
98     \RequirePackage{ragged2e}%
99     \renewcommand*{\ila@RaggedRight}{\RaggedRight}%
100   }{%
101     \PackageWarning{idxlayout}{%
102       Package ragged2e not available, therefore\MessageBreak
103       substituting command raggedright for RaggedRight\MessageBreak
104     }%
105     \renewcommand*{\ila@RaggedRight}{\raggedright}%
106   }%
107 }
108 \ila@defradiokey{justific}{ju}
109 \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 `noteseq` as vertical space. `\noindexprenote` defines `\ila@prenote` as `\relax` (which also means that the control sequence test `\ifundef` will yield true).

```

110 \newcommand{\ila@prenote}{}
111 \newcommand{\setindexprenote}[1]{%
112   \def\ila@prenote{%
113     \begingroup#1\par\nobreak\endgroup
114     \vspace{\ila@noteseq}%
115   }%

```

```

116 }
117 \newcommand*{\noindexprenote}{\let\ila@prenote\relax}
118 \noindexprenote

```

`\indexstheadcase` The `\indexstheadcase` command allows to avoid hard-coding `\MakeUppercase` into the index header definition (as is currently done by the standard classes).

```

119 \newcommand*{\indexstheadcase}{\MakeUppercase}

```

3.3 Emulating specific document classes and packages

`\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.

```

120 \newcommand*{\ila@classtype}{0}
121 \@ifclassloaded{memoir}{%
122   \def\ila@classtype{2}%
123 }{%
124   \ifundef{\KOMAClassName}{%
125     }{%
126       \def\ila@classtype{1}%
127     }%
128 }

```

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`.

```

129 \ifcase\ila@classtype\relax
130   \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`.

```

131 \def\ila@prologue{%
132   \ifundef{\chapter}{%
133     \section*{\indexname}%
134     \ifbool{ila@totoc}{%
135       \addcontentsline{toc}{section}{\indexname}%
136     }{%
137       }%
138     }{%
139       \chapter*{\indexname}%

```

```

140     \ifbool{ila@totoc}{%
141         \addcontentsline{toc}{chapter}{\indexname}%
142     }{%
143     }%
144 }%
145 \mkboth{\indexstheadcase\indexname}{\indexstheadcase\indexname}%
146 }

```

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`. We also provide `\MakeMarkcase` to ensure compatibility with older versions (below 3.10) of KOMA-Script.

```

147 \or
148 \def\ila@tc@true{\KOMAoptions{index=totoc}}%
149 \def\ila@tc@false{\KOMAoptions{index=nottotoc}}%
150 \ila@defradiokey[true]{totoc}{tc}%
151 \providecommand{\MakeMarkcase}[1]{#1}%

```

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 and also (for the moment) do not use `\index@preamble`.

```

152 \def\ila@prologue{%
153     \ifundef{\chapter}{%
154     }{%
155         \ifundef{\index@preamble}{%
156         }{%
157             \setchapterpreamble{\index@preamble}%
158         }%
159     }%
160     \idx@@heading{\indexname}%
161     \mkboth{\MakeMarkcase{\indexname}}{\MakeMarkcase{\indexname}}%
162     \ifundef{\chapter}{%
163     }{%
164         \thispagestyle{\indexpagestyle}%
165     }%
166 }

```

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

```

167 \or
168 \def\ila@tc@true{\boolfalse{noindexintoc}}%
169 \def\ila@tc@false{\booltrue{noindexintoc}}%
170 \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`.

```

171 \def\ila@prologue{%
172   \chapter*{\indexname}%
173   \ifbool{noindexintoc}{%
174     }{%
175       \addcontentsline{toc}{chapter}{\indexname}%
176     }%
177   \ifbool{artopt}{%
178     }{%
179       \thispagestyle{indextitlepagestyle}%
180     }%
181   \indexmark
182   \preindexhook
183 }
184 \fi

```

`\ila@packindadjust` If the index package is loaded, the command `\ila@packindadjust` will – in accordance with the settings for individual indexes – redefine `\indexname` and (depending on the state of `\index@prologue` which stores the second optional argument of `\printindex`) reset the argument of `\setindexprenote`. Because the `memoir` class (with built-in features similar to those of the `index` package) feigns the presence of the package, `\ila@packindadjust` is also let to `\relax` if the reported package date is newer than that of the last official version (v4.2beta).

```

185 \ifpackageloaded{index}{%
186   \@ifpackagelater{index}{2004/01/21}{%
187     \let\ila@packindadjust\relax
188   }{%
189     \def\ila@packindadjust{%
190       \edef\indexname{\the\@nameuse{idxtitle@\@indextype}}%
191       \ifdefempty{\index@prologue}{%
192         }{%
193           \setindexprenote{\index@prologue}%
194         }%
195       }%
196     }%
197   }{%
198     \let\ila@packindadjust\relax
199   }

```

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

```

200 \ProcessKeyvalOptions*

```

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

```

201 \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.

```
202 \renewenvironment{theindex}{%
203   \ifbool{@twocolumn}{%
204     \boolfalse{@restonecol}%
205     \onecolumn
206   }{%
207     \booltrue{@restonecol}%
208   }%
```

\multicolsep is set to 0 pt, 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. Note that in order to be effective, this has to be done before opening a **multicols** environment.

```
209   \setlength{\multicolsep}{\z@}%
210   \setlength{\columnsep}{\indexcolsep}%
211   \setlength{\columnseprule}{\indexrule}%
212   \ila@packindadjust
```

The necessary adjustments to the index package are carried out.

\ila@prologueplus If a single-column index is requested, we simply execute **\ila@prologue**, switch to **\indexfont** and (if the boolean option **columnnote** is disabled) 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** (or, if the boolean option **unbalanced** is enabled, **multicols***) environment and specify the value of **minspace** as its second optional argument.

```
213   \def\ila@prologueplus{%
214     \ila@prologue
215     \indexfont
216     \ifbool{ila@columnnote}{%
217       }{%
218         \ila@prenote
219       }%
220     }%
221     \ifnumcomp{\theidxcols}{<}{\tw@}{%
222       \ila@prologueplus
223     }{%
224       \ifbool{ila@unbalanced}{%
225         \begin{multicols*}{\theidxcols}[\ila@prologueplus][\ila@minspace]%
226       }{%
227         \begin{multicols}{\theidxcols}[\ila@prologueplus][\ila@minspace]%
228       }%
229     }%
```

We set **\parskip** to 0 pt plus 0.3 pt (exactly as in the standard classes) and switch to the proper justification. For the **scartcl** class, we append its index preamble.

We also typeset a possible prenote in case `columnnote` is enabled. Last, but not least, we redefine `\item`.

```

230 \setlength{\parskip}{\z@ \@plus 0.3\p@}%
231 \indexjustific
232 \ifundef{\chapter}{%
233   \ifundef{\index@preamble}{%
234     }{%
235       \index@preamble\par\nobreak
236     }%
237   }{%
238   }%
239 \ifbool{ila@columnnote}{%
240   \ila@prenote
241 }{%
242 }%
243 \let\item\@idxitem

```

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

```

244 }{%
245 \ifnumcomp{\theidxcols}{<}{\tw@}{%
246 }{%
247   \ifbool{ila@unbalanced}{%
248     \end{multicols*}%
249   }{%
250     \end{multicols}%
251   }%
252 }%
253 \ifbool{@restonecol}{%
254 }{%
255   \twocolumn
256 }%
257 }

```

The end of the package.

```

258 \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”. *TUGboat* 30.1 (2009), pp. 110–122. <http://www.tug.org/TUGboat/tb30-1/tb94wright-keyval.pdf> (accessed March 30, 2012).

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@indentunit</code> 33,
<code>\@idxitem</code> 39, 44, 45, 243	environments:	41–43, 48–50, 64
<code>\@ifclassloaded</code> 17, 121	<code>theindex</code> 202	<code>\ila@initsep</code> . . . 70, 75
<code>\@ifpackagelater</code> . 186	F	<code>\ila@it@abshang</code> . . . 40
<code>\@ifpackageloaded</code> 185	font (option) 4	<code>\ila@it@relhang</code> . . . 47
<code>\@indextype</code> 190	<code>\footnotesize</code> 81	<code>\ila@it@singlepar</code> . 63
<code>\@mkboth</code> 145, 161	H	<code>\ila@ju@RaggedRight</code> 93
A	<code>\hangindent</code> . 39, 53, 58	<code>\ila@ju@raggedright</code> 91
<code>\addcontentsline</code> .	hangindent (option) . . 3	<code>\ila@ju@standard</code> . . 85
. . . . 135, 141, 175	I	<code>\ila@minspace</code>
<code>\appto</code> 22, 23	<code>\idx@@heading</code> 160	 70, 225, 227
<code>\AtEndPreamble</code> 96	<code>\idxlayout</code> 4, 201	<code>\ila@notesep</code> . . 70, 114
C	<code>\ifdefempty</code> 191	<code>\ila@packindadjust</code>
<code>\chapter</code> . 132, 139,	<code>\IfFileExists</code> 97	 185, 212
153, 162, 172, 232	<code>\ifnumcomp</code> . . 221, 245	<code>\ila@prenote</code>
<code>columnnote</code> (option) . 4	<code>\ifundef</code>	 110, 218, 240
<code>columns</code> (option) 2	124, 132, 153,	<code>\ila@prologue</code>
<code>\columnsep</code> 210	155, 162, 232, 233	131, 152, 171, 214
<code>columnsep</code> (option) . . 3	<code>\ila@classtype</code> 120, 129	<code>\ila@prologueplus</code> 213
<code>\columnseprule</code> . . . 211	<code>\ila@defradiokey</code> .	<code>\ila@RaggedRight</code> .
D	 5, 68,	. . . 92, 94, 99, 105
<code>\DeclareBoolOption</code>	82, 108, 150, 170	<code>\ila@subindent</code> . 36,
. 32, 76, 130	<code>\ila@fo@current</code> . . . 78	42, 44, 49, 53, 54
<code>\DeclareStringOption</code>	<code>\ila@fo@footnotesize</code>	<code>\ila@subsubindent</code>
. . . . 36–38, 73, 74	 81	 36,
<code>\define@key</code> 6,	<code>\ila@fo@normalsize</code> . 79	43, 45, 50, 58, 59
15, 30, 31, 34, 71	<code>\ila@fo@small</code> 80	<code>\ila@tc@false</code> 149, 169
<code>\deflength</code> 53, 58	<code>\ila@hangindent</code> . .	<code>\ila@tc@true</code> . 148, 168
	 36, 39,	indentunit (option) . . 3
	41, 48, 53, 58, 64	<code>\index@preamble</code> . .
		155, 157, 233, 235

<code>\index@prologue</code>	191, 193	<code>\normalsize</code>	79	R	
<code>\indexcolsep</code>	.. 24, 210	<code>notesep</code> (option)	4	<code>\RaggedRight</code>	99
<code>\indexfont</code>	.. 4, 77, 215	O		<code>\raggedright</code>	.. 91, 105
<code>\indexjustific</code>	4, 84, 231	<code>\onecolindex</code>	14	<code>rule</code> (option)	3
<code>\indexmark</code>	 181	<code>\onecolumn</code>	205	S	
<code>\indexname</code>		options:		<code>\section</code>	133
	133, 135, 139,	<code>columnnote</code>	4	<code>\setchapterpreamble</code>	157
	141, 145, 160,	<code>columns</code>	2	<code>\setindexprenote</code>	 4, 110, 193
	161, 172, 175, 190	<code>columnsep</code>	3	<code>\setkeys</code>	 16,
<code>\indexpagestyle</code>	.. 164	<code>font</code>	4		19, 22, 23, 35,
<code>\indexrule</code>	.. 24, 211	<code>hangindent</code>	3		69, 72, 83, 109, 201
<code>\indexspace</code>	 75	<code>indentunit</code>	3	<code>\SetupKeyvalOptions</code>	.. 4
<code>\indexstheadcase</code>	 4, 119, 145	<code>initsep</code>	4	<code>\small</code>	 80
<code>\indexsubsdelim</code>	.. 4, 62	<code>itemlayout</code>	3	<code>subindent</code> (option)	.. 3
<code>initsep</code> (option)	 4	<code>justific</code>	4	<code>\subitem</code>	.. 40, 51, 65
<code>itemlayout</code> (option)	.. 3	<code>minspace</code>	4	<code>subsubindent</code> (option)	3
J		<code>notesep</code>	4	<code>\subsubitem</code>	.. 40, 56, 66
<code>justific</code> (option)	 4	<code>rule</code>	3	T	
K		<code>subindent</code>	3	<code>\theidxcols</code>	 14,
<code>\KOMAClassName</code>	.. 124	<code>subsubindent</code>	3		221, 225, 227, 245
<code>\KOMAOptions</code>	.. 148, 149	<code>totoc</code>	4	<code>theindex</code> (environ-	
M		<code>unbalanced</code>	2	ment)	 202
<code>\MakeMarkcase</code>	151, 161	P		<code>\thispagestyle</code>	164, 179
<code>\MakeUppercase</code>	.. 119	<code>\PackageError</code>	10	<code>totoc</code> (option)	 4
<code>minspace</code> (option)	.. 4	<code>\PackageWarning</code>	.. 101	<code>\twocolindex</code>	 14
<code>\multicolsep</code>	 209	<code>\parfillskip</code>	88	<code>\twocolumn</code>	 255
N		<code>\parindent</code>	87	U	
<code>\noindexprenote</code>	4, 110	<code>\parskip</code>	230	<code>unbalanced</code> (option)	.. 2
		<code>\preindexhook</code>	182	<code>\unskip</code>	 65, 66
		<code>\ProcessKeyvalOptions</code>	 200		

Change History

v0.4	General: Initial public release	 1	v0.4c	General: Corrected option <code>justific=RaggedRight</code>	 4
v0.4a	General: New option <code>unbalanced</code>	.. 2		Package <code>ragged2e</code> now loaded automatically	 4
v0.4b	General: Corrected emulation of <code>\setindexpreamble</code>	 5	v0.4d	General: Adapted emulation of <code>theindex</code> to KOMA-Script v3.10	.. 5
	New option <code>columnnote</code>	 4		Added <code>\indexstheadcase</code>	 4
	Package index now supported	.. 5			