

The **interfaces*** package

interfaces with keys for other packages (**FC**)

<florent.chervet at free.fr>

2010/11/16 – version 2.2 – pgf v2.1 / pgf v2.0 compatible

Abstract

interfaces provides a small number of user friendly macros to access features implemented in frequently used packages. Most of the macros are base on **pgfkeys**, hence the “key=value” syntax.

interfaces is mainly based on **pgfkeys**¹ and **scrlfile** for its different modules are loaded \AfterPackage. ε -TeX is required of course... Each interface is actually made as a separate module, which is loaded automatically after its *master package*. Adding \usepackage{**interfaces**} into your document preamble will make all modules active (*ie*. they will be loaded or not, depending on your document preamble). If you prefer not to use all modules, you can load them separately as a package, for example, type: \usepackage[**interfaces-titlesec**]{**interfaces**} *after* \usepackage{**titlesec**}. A better way is to add \usepackage[**interface-base**] to your preamble, and then: \LoadInterface{**titlesec,tocloft**} indifferently before or after \usepackage{**titlesec,tocloft**}. Another possibility to restrict the modules is to give them as options for the main package: \usepackage[**titlesec,tocloft**]{**interfaces**}.

interfaces-base is a mandatory “module”. It loads **interfaces-scrlfile**, **interfaces-etoolbox**² and **interfaces-pgfkeys**: those are the *system-layer* of **interfaces**.

~~~~~ Example ~~~~~

```
\pagestyle[document]{%
    rules=.5pt,
    foot/left/odd=\arabic{page},
    foot/right/even=\arabic{page}/\pageref{LastPage},
    head/font=\color[gray]{.85}\bfseries\small,
    head/left/even=\leftmark,
    head/left=\thefirstmarks{MiniSection},% interfaces-marks
}
\pagestyle[plain]{% for chapters...
    pagestyle=document,% preloaded settings
    head=,
    foot/left/odd = \roman{page},
    norules, head/rule=.4pt
    head/center=This is the title,
}
\sectionformat[section]{hang]{%
    label=\arabic{section},
    labelsep=,
    font=\bfseries\Large,
    bookmark={bold,color=blue},% interfaces-bookmark
    top=6pt plus3pt minus2pt,
    bottom=3pt plus2pt,
    break=\marksthecs{section}{lastsectiontitle},% interfaces-hypbmsec
    mark=\marksthe{section}{#1},% interfaces-marks
}
\sectionformat[subsection]{hang}{%
    label=, labelsep=,
    font=\bfseries,
    bookmark=italic,
    top*=1.2,% a 120% scaling
    bottom+=0pt plus4pt,% add to previous value
    mark=\markright{#1},
}
\bottomtitles=.2\textheight
\tocsetup{%
    depth=3, title=Contents, pagenumbers=on,
    pagestyle=plain,% first page style: tocloftpagestyle
    title/top=1.5ex plus.5ex,
    title/bottom=0pt,
    section/indent=1em,
    section/number/width=1.5em,
    section/font+=\larger\bfseries,% append to previous setting
    section/leaders,section/dotsep,
    subsection/indent=2em,
    subsection/number/width=2.5em,
    subsubsection/font=\scriptsize,
    subsection/leaders,subsection/dotsep,% default values
    subsubsection/pagenumbers=off,% no page numbers
    subsubsection/dotsep=% no leader
    columns=2,% same as twocolumns
    columns/sep=.5em,
    columns/rule/width=1pt,columns/rule/color=blue,
    label=toc,% gives a label to the t.o.c.
    bookmark={text=Contents,bold},% interfaces-bookmark
}
\tableofcontents
\mainmatter \pagestyle{document}
```

This documentation is produced with the +DocStrip+ utility.

- To get the documentation, run (thrice): pdflatex interfaces.dtx
- To get the index, run: makeindex -s gind.ist interfaces.idx
- To get the package, run: etex interfaces.dtx

The .dtx file is embedded into this pdf file thank to embedfile by H. Oberdiek.

1. **pgfkeys**: CTAN:graphics/pgf/base/generic/pgf/utilities

2. All macros in **interfaces-etoolbox** are defined with \providecommand except \removefromstart which is used by \repeatcell.

Contents

1 Interface L^AT_EX	3	21 Implementation	25
2 Interface TikZ	3	21.1 Interfaces	25
3 Interface titlesec: \sectionformat	4	21.1.1 Requirements	
4 Interface hyperref	6	21.1.2 The package options / list of interfaces	
5 Interface bookmark	6	21.2 Interfaces-base	25
6 Interface hypbmsec	7	21.2.1 Bug Fixes	
7 Interface tocloft: \tocsetup	8	21.2.2 Interfaces patches	
8 Interface fancyhdr: \pagesetup	9	21.2.3 Helper macros	
9 Interface marks	10	21.2.4 Numbers	
10 Interface makecell: \repeatcell	11	21.2.5 Units and math units	
10.1 Conversion to scaled points	12	21.2.6 Default color	
10.2 Variable length cell	12	21.3 Interface L ^A T _E X	30
10.3 Empty rows	12	21.3.1 \changefont	
10.4 color palette	13	21.4 Interface TikZ	31
10.5 tabularcalc-like tables (with dcolumn package) and pgfmath	13	21.4.1 Catcode verifications	
10.6 Double entry array: the Body Mass Index	14	21.4.2 List of pgf/TikZ packages and libraries	
10.7		21.4.3 pgfkeys definitions for \usetikz	
11 Interface enumitem	15	21.5 Interface titlesec: \sectionformat	34
12 Interface truncate	16	21.5.1 titlesec modifications	
13 Interface appendix	17	21.5.2 \sectionformat	
14 Interface embedfile: \embedthisfile	17	21.5.3 Keys for \sectionformat	
15 Interface environ	17	21.5.4 \bottomtitles	
16 Interface umrand	17	21.6 Interface hyperref	40
17 Interface pgfkeys	18	21.7 Interface bookmark	41
17.1 /.blank	18	21.7.1 star keys* definitions	
17.2 /.is private	19	21.7.2 ref and pageref keys	
17.3 /.store in cs, /.estore in cs	19	21.8 Interface hypbmsec	44
17.4 Boolean keys: /.true, /.false, /.on /.off	19	21.8.1 interfaces-hypbmsec	
17.5 /.is multiple choice	19	21.8.2 interfaces-hypbmsec.code.tex	
17.6 /.is enum	20	21.9 Interface fancyhdr: \pagesetup	46
17.7 /.store as code	20	21.9.1 \pagesetup	
17.8 /.store as code in cs	20	21.9.2 Keys for \pagesetup	
17.9 /.store as color	20	21.10 marks	52
17.10 /.store as color in cs	21	21.11 Interface tocloft: \tocsetup	54
17.11 /.store as number	21	21.11.1 Specific handlers for tocloft	
17.12 /.store as number in cs	21	21.12 Interface makecell: \repeatcell	60
17.13 /.store as glue	21	21.12.1 countvalues	
17.14 /.store as dimen	21	21.12.2 \repeatcell	
17.15 /.store as glue in cs	21	21.12.3 colortbl: patch for \CT@cellc	
17.16 /.store as dimen in cs	22	21.13 Interface enumitem	66
17.17 /.estore as dimen in cs	22	21.14 Interface truncate	67
18 Interface etoolbox	22	21.15 interface fancyrb	69
18.1 New macros provided with \providetcommand	22	21.16 Interface appendix	69
19 Interface scrfile	22	21.17 Interface embedfile: \embedthisfile	70
19.1 Star forms for \BeforeFile and \AfterFile	22	21.18 Interface environ	71
19.2 Allowing \BeforeFile and \AfterFile when \@input is invoked	23	21.19 Interface umrand	71
20 List of corrected or patched commands	23	21.20 Interface scrfile	71
20.1 LaTeX	23	21.21 Interface pgfkeys	72
20.2 bookmark	23	21.21.1 interfaces-pgfkeys	
20.3 colortbl	24	21.22 Interface etoolbox	85
20.4 hypbmsec	24	22 History	87
20.5 enumitem	24	[2010/11/16 v2.2]	87
20.6 pgfkeys	24	[2010/11/15 v2.1]	87
		[2010/10/31 v2.0]	87
		[2010/10/27 v1.2]	87
		[2010/10/20 v1.0]	87
23 References	24	24 Index	89
README	24		

1 Interface LATEX

```
\changefont{ encoding |enc = changes \fontencoding
            family |fam = changes \fontfamily
            series |ser = changes \fontseries
            shape |sha = changes \fontshape
            size |size* |size+ |size- = changes the font size (default unit is pt)
baselineskip | baselineskip* + - | skip = changes the \baselineskip for the font
linespread | linespread* + - | spread = changes the line spread (a real value) }
```

Example: adds 5pt to the font size (pt is added if not specified) and scale the linespread by 130%:

`\changefont{ size+ = 5, spread* = 1.3}` (\selectfont is included in the macro).

If a key is not specified, the current value remains.

Finally, \changefont is compatible with hyperref thank to \pdfstringdefDisableCommands.

2 Interface TikZ ([pgfmanual](#))

```
\usetikz { basic=..., loads tikz with libraries calc, arrows, scopes and backgrounds
           all=..., loads tikz with all packages, pgf-modules and TikZ-libraries
           any other key=..., loads tikz and the specified package, module or library
                         using \usepackage, \usepgfmodule or \usetikzlibrary }
```

`\usetikz` is a facility provided to load pgf, tikz, pgf-modules, and tikz-libraries, without the need to know if you must use \usepackage or \usepgfmodule or \usetikzlibrary.

`\usetikz` does not parse options. If you are used to load pgf with option (f.ex. `version=0.96`), please use the syntax:

`\PassOptionsToPackage{version=0.96}{pgf}`

~~~~~ MEMO ~~~~~			
TikZ/pgf package	pgfmodules	TikZ libraries	
pgf	matrix	arrows	decorations.pathreplacing
pgfmath	plot	chains	decorations.shapes
pgfcalendar		er	decorations.text
pgfplots		scopes	decorations.markings
pgfpages		snakes	decorations.fractals
tikz		shapes	decorations.footprints
		shapes.misic	positioning,trees
		shapes.geometric	automata
		shapes.arrows	patterns
		shapes.multipart	fadings
		shapes.callouts	shadows
		shapes.symbols	through
		shapes.gates.logic.US	backgrounds
		shapes.gates.logic.IEC	petri
		decorations	fit
		decorations.pathmorphing	mindmap
			folding
			plothandlers
			plotmarks
			calc
			matrix

sorry for the blank cell, but I didn't get out with colortbl and the \multicolumn entry...

**3 Interface ↗ titlesec : \sectionformat**

```
\sectionformat{\section} { class=..., page|top|straight [\sup-level]
    shape=..., hang|block|display|runin|leftmargin|rightmargin|
        drop|wrap|frame|runinifempty
    font|format=..., label=..., labelsep|sep=...
        left=..., right=..., top=..., bottom=...
    before=..., after=...
    mark=..., break=..., thesection=..., }
```

`\sectionformat` is a pgfkeys-command for replacement of both `\titleformat` and `\titlespacing`. If a key is not given, its value remains the same while with `\titleformat` or `\titlespacing`, you must specify all values in a rigid order... Moreover, `\sectionformat` can be used concurrently with `\titleformat` and `\titlespacing` (although `\sectionformat` is deemed more comfortable, this allows to quickly modify some files written without interfaces.)

Keys corresponding to TeX code (`font`, `before` or `after`) are associated with starred keys and `+ keys`: they append their content to the previous value. Ex: `font+={\larger}` adds `\larger` to the previous settings.

For dimensions or skips, the **+** key adds, the **-** key subtract and the ***** key scales.

MEMO		
class	name of the class	<p><code>page</code>=like standard \part  <code>top</code>=like standard \chapter  <code>straight</code>=like standard \section etc.  example: \sectionformat\subchapter{class=straight[\chapter]}</p>
shape	shape	<p><code>hang</code>=hanging label (like \section)  <code>block</code>=paragraph without formatting (for centered titles)  <code>display</code>=like standard \chapter  <code>runin</code>=like standard \paragraph  <code>leftmargin</code>=left margin  <code>rightmargin</code>=right margin  <code>drop</code>=wraps the text around the title (the first paragraph should be longer than the title, otherwise, they overlap)  <code>wrap</code>=like <code>drop</code> but the space reserved for the title is variable  <code>frame</code>=like <code>display</code>+puts the title in frame  <code>runinifempty</code>=<code>hang</code> but no label if the title is empty</p>
<code>font</code> = <code>format</code>	format for title	applied to the label and the title
<code>label</code>	title label	ex: \arabic{section}
<code>sep</code> ≡ <code>labelsep</code>	space between label and title	
<code>left</code>	left margin	a glue: horizontal space before
<code>right</code>	right margin	a glue: horizontal space after
<code>top</code>	vertical space before	a glue: vertical space before
<code>bottom</code>	vertical space after	a glue: vertical space after
<code>mark</code>	macro with 1 parameter	ex \sectionmark: redefines \...mark#1... (the argument is the title)
<code>break</code>	parameterless macro	ex \sectionbreak to be executed just before the section command
<code>thesection</code>	toc and ref labels	redefines \thesection \thesection, \thesubsection etc.. are used to print references and the label in the table of contents
<code>before</code>	code before title body	
<code>after</code>	code after title body	

If the bookmark package is loaded, with `interfaces-bookmark`, the following keys are defined for each section:

<b>bookmark/bold</b>	bold bookmark text	the text of the bookmark for that sectionning command will be <b>bold</b>
<b>bookmark/italic</b>	italic bookmark text	the text of the bookmark for that sectionning command will be in <i>italic</i>
<b>bookmark/color</b>	colored bookmark text	the text of the bookmark for that sectionning command will be <b>colorized</b>
<b>bookmark/off</b>	disable the bookmark	no bookmark for that sectionning command alternatively you may set <b>bookmark/level</b> to a huge value...
<b>bookmark/style</b>	use a <i>style</i>	the <i>style</i> must be previously declared with \bookmarkdefinestyle
<b>bookmark/level</b>	level of the bookmark	defines \toclevel@{sectionname}, see bookmark documentation

The syntax: `\sectionformat\subsection{bookmark={bold,italic,color=green,on,view=FitB}}` also works.

→ Key bookmark is an example of a key defined with the handler `./is` multiple choice.

The starred keys **`bold*`**, **`italic*`** and **`color*`** are also provided. For example:

```
\sectionformat\section{
    after=\sectionformat\subsection{bookmark/color*=red}
}
```

makes the bookmarks corresponding to the first subsection in each section to be red. If you ascertain that there is no bookmark between the one corresponding to the section and the one corresponding to its first subsection, then this is the same as: “`after=\bookmarksetup[color*=red]`”.

EXAMPLE
<pre>\sectionformat\section[hang]{%     font=\bfseries\scshape,     label=\Roman{chapter}\ \textbullet\ \arabic{section},     bottom=1ex plus.4ex minus.2ex,     top+=3pt,     mark=\markright{\#1},% =&gt; this defines \def\sectionmark{\markright{\#1}}     break=\goodbreak%=&gt; this defines \def\sectionbreak{\goodbreak}     thesection=\Roman{chapter}.\arabic{section} } % redefines \def\thesection{...}</pre>

`titlesec` provides *extended settings* for numberless (*i.e.* starred) sections with the syntax:

```
\titleformat{name=\section, numberless}{...}
```

One swallow does not a summer make, `interface-titlesec` does not provide this feature in a `key=value` stuff but just type:

```
\sectionformat\section*{ key=value }
```

to give to your `\section*` command a special format. Keys `label` and `labelsep` are not ignored, this means that `\section*` can be defined in a way it produces a numbered section! Albeit, this would be weird because the `\section` counter is not updated for starred sections...

Presently, `\sectionformat` does not implement the *extended settings* of `titlesec` for odd and even pages.

However, it is always possible to use `\sectionformat` concurrently with `\titleformat` and `\titlespacing`. A better way to use different settings for odd and even pages is the use of `\ifthispageodd` from the KOMA-Script classes, or the features provided by `changepage.sty`, which is almost the same: actually `titlesec` test oddity of the `page` counter (*i.e.* `\TeX\count` register nr 0) while oddity of `\pageref` should be tested... Hence the decision I came not to implement different settings for odd and even pages in `\sectionformat`.

<code>\bottomtitles{&lt;glue percentage of \textheight&gt;}</code>
--------------------------------------------------------------------

This sets the value of `\bottomtitlespace` (see `titlesec` documentation):

<code>\bottomtitles{.2}</code>	set <code>\bottomtitlespace</code> to <code>.2\textheight</code>
<code>\bottomtitles=.2\textheight</code>	is the same

You can set a negative value: `\bottomtitles = -1` to allow bottom titles.

## 4 Interface ⇔ hyperref

```
\raisedhyperdef[<raised dimen>]{<category>}{<name>}{<text>}
```

Package `hyperref` provides `\hyperdef` and `\Hy@raisedlink` but no easy way to insert a “raised link” with `\hyperdef` (or `\hypertarget`).

`\raisedhyperdef` can be used exactly like `\hyperdef`, but the pdf-anchor will be raised of the specified dimension.

## 5 Interface ⇔ bookmark

A few new keys appear with `interfaces` for the command `\bookmarksetup`:

<code>\bookmarksetup {</code>	<code>bold*=...,</code> will make the bookmark text bold, but only for the very next bookmark <code>italic*=...,</code> <i>idem</i> for italic <code>color*=...,</code> <i>idem</i> for color: the previous color will be restored after the next bookmark <code>disable=...,</code> will disable the bookmarks (the same as <code>\bookmarksetup{depth=-2}</code> ) <code>disable*=...,</code> will disable only the very next bookmark	<code>}</code>
-------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------

Furthermore the following code:

```
\sectionformat\section{bookmark={bold,color=blue}}
\bookmarksetup{color*=red}
\section{A lovely red, bold bookmark here!}
\section{And the following bookmark is blue and bold}
```

will produce a bookmark with attributes: `color=red, bold=true`. In other words, the starred keys of `\bookmarksetup` overwrite the “default” settings of `\sectionformat` for the very next bookmark.

*Note that this code requires `titlesec`, `bookmark`, `interfaces-titlesec` and `interfaces-bookmark`.*

Three new keys are added to the `\bookmark` command:

<code>\bookmark{</code>	<code>pageref</code> =the destination of the bookmark is the page referenced <code>ref</code> =the destination of the bookmark is the anchor associated to the reference <code>atend</code> =the bookmark will be inserted at the end of the document	<code>}</code>
-------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------

I admit those keys will not be used very often... but imagine you want to put a bookmark at the end, which points to the table of contents, then:

```
\bookmark[ref=TableOfContents,atend]{Back To Contents}
```

Provided that the label `TableOfContents` has been defined, for example with `\tocsetup`, if `tocloft` and `interfaces-tocloft` are loaded:

```
\tocsetup{label=TableOfContents} \tableofcontents
```

The `atend` key requires of course that the package `bookmark` was loaded with the `[atend]` option (otherwise, `bookmark.sty` gives a warning).

## 6 Interface ☈ `hypbmsec`

The features provided by `interfaces-hypbmsec` requires `interfaces-titlesec`, and therefore the `titlesec` package: when using `interfaces` with automatic module loading, (or while invoking `\LoadInterface{hypbmsec}`), this module is loaded only after both `titlesec.sty` and `hypbmsec.sty`.

Package `hypbmsec` provides extended sectionning commands with the syntax:

`\section(out)[toc]{title}`      or      `\section[toc](out){title}`

(`out`) is used for the bookmark (outline)

Where:    `[toc]` for the table and contents and the marks

`{title}` to typeset the section title in the document

Internally, `hypbmsec` defines `\hbs@bmstring` (the string used for the bookmark entry) and `\hbs@tocstring` (the string used for the table of contents and the marks).

`interfaces` takes advantage of this and defines:

<code>\lastsectiontitle</code>	expands to the title of the last <code>\section</code>
<code>\lastsectionbookmark</code>	expands to the text of the bookmark for the last <code>\section</code>
<code>\lastsectiontoc</code>	expands to the entry of the last <code>\section</code> in the table of contents

Macros for each sectionning command are defined:

<code>\lastparttitle</code>	<code>\lastchaptertitle</code>	
<code>\lastsectiontitle</code>	<code>\lastsubsectiontitle</code>	<code>\lastsubsubsectiontitle</code>
<code>\lastparagraphtitle</code>	<code>\last subparagraphtitle</code>	

with their `...bookmark` and `...toc` counterparts.

Assignment to those macros is automatic. More precisely, `\lastsectiontitle` is defined **after** `\sectionbreak` and **before** `\sectionmark`: this means that in the fields `after` or `mark` of  $\sim\backslash\text{sectionformat}$ , `\lastsectiontitle` has been updated yet. While in the fields `before`, and `break`, `\lastsectiontitle` is not yet updated. (For chapters, this is the same if `titlesec` formats the chapter titles, *i.e.* if `\sectionformat\chapter` or `\titleformat\chapter` has been invoked. Otherwise, `\lastchaptertitle`, `\lastchapterbookmark` and `\lastchaptertosoc` are set at `\stepcounter{chapter}`).).

Values are automatically reset to `\relax` when a section of a superior level is created.

This feature has been implemented essentially to make the use of a `\mark` command at `\sectionbreak` easier. Mark at `\sectionbreak` is important to get the correct behaviour for marks described in `interface-marks` (page 10 under  $\sim\backslash\text{ifmarksequal}$ ). With this feature, when a new `\section` is created, `\lastsubsection` is reset to `\relax` therefore, `\marksthecs{\subsection}{\lastsubsection}` will not mark anything.

Setting the keys overwrites the previous value, unless the `+` key (append code/add for dimensions), the `-` key (subtract) or the starred-key`*` (scale dimensions) is used. `\tocsetup` settings apply to `\tableofcontents` and also to `\shorttableofcontents` (package `shorttoc`). `\tocsetup` can be invoked more than once in a document...

<code>\tocsetup{</code>	<code>depth = \setcounter{tocdepth}</code> This is a L ^A T _E X rather than tocloft's ... However with <code>interfaces</code> you can change the t.o.c. depth anywhere inside your document, just like bookmarks depths (two runs are required)
<code>title contentsname = redefines \contentsname</code>	
<code>title/top title/top+ title/top- title/top* = vertical space before the title (set add subtract scale)</code>	
<code>title/bottom title/bottom+ etc. = vertical space after the title</code>	
<code>title/font title/font* = the title font (set append code)</code>	
<code>title/after title/after* = hook to execute just after the title (\cftaftertoctitle...)</code>	
<code>parskip parskip+ etc. parsep = modifies (or add to) \cftparskip</code>	
<code>dotsep = modifies \cftdotsep</code> if no dimen is specified, <i>mu</i> is assumed, otherwise, conversion to $\mu \approx 1em/18$ is supplied. Default value is <i>4.5mu</i> and blank value is the same as <code>\cftnodots</code>	
<code>dot = modifies \cftdot, the dot used for leaders produced by \cftdotfill</code>	
<code>pagestyle = page style for the first page (\tocloftpagestyle)</code>	
<code>mark = the mark command: \cftmarktoc</code>	
<code>example: \chaptermark\contentsname</code>	
<code>before before* after after* = hooks to execute before or after the command \tableofcontents</code>	
<code>twocolumns columns multicol = twocolumns=true is the same as columns=2</code> Makes a multicolumn table of contents. Package <code>multicol</code> is required!	
<code>columns/sep columns/sep+ columnsep etc. = sets (or add) a &lt;dimen&gt; to \columnsep</code>	
<code>columns/rule columns/rule+ columnseprule etc. = sets (or add) a &lt;dimen&gt; to \columnseprule</code>	
<code>columns/color columnseprulecolor = modifies \columnseprulecolor</code>	
<code>multicols multicols+ etc. = modifies (or add) a &lt;glue&gt; to \multicols</code> <code>\multicols is the space added before and after <code>multicol</code> environment. Its default value is <code>0pt plus 4pt minus 3pt</code>.</code>	
<code>multicols/begin twocolumns/before = hooks just after \begin{multicols} ...</code>	
<code>multicols/end twocolumns/after = ... and before \end{multicols}</code>	
<code>label = the label for reference to the table of contents</code>	
<code>ref = the reference printed by \ref when referencing the table of contents</code>	
<code>(default is \contentsname. In any case, \nameref prints \contentsname.)</code>	
<code>bookmark/text = the bookmark text for the table of contents (if empty, no bookmark is set)</code>	
<code>bookmark/bold bookmark/italic bookmark/color = the bookmark style for the table of contents</code>	
<code>bookmark/level = the bookmark level: either a number or section or chapter</code>	
<code>Default is chapter, or section if the class does not provide chapters.</code>	}

`bookmark` key and sub-keys are available only if the package `bookmark` is loaded with `interfaces-bookmark`.

The following keys apply to each section levels. Section level names are : `part` / `chap` or `chapter` / `sec` or `section` / `subsec` or `subsection` / `subsubsec` or `subsubsection` / `para` or `paragraph` / `subpara` or `subparagraph` / `fig` or `figure` / `subfig` or `subfigure` / `tab` or `table` / `subtab` or `subtable`. ¶ may be one of those names (see `tocloft` documentation).

<code>\tocsetup{</code>	
<code>¶/top ¶/top+ ¶/skip ¶/beforeskip = modifies (or add to) the vertical &lt;glue&gt; before an entry</code>	
<code>¶/indent ¶/indent+ ¶/indent- ¶/indent* = modifies (or add etc.) the indentation of the entry</code>	
<code>¶/font ¶/font* = the font of the ¶ section title</code>	
<code>¶/number/width number/width+ numwidth = modifies the space allowed to typeset the title numbers</code>	
<code>¶/number/before number/before* presnum = code before the section number (see pseudo code below)</code>	
<code>¶/number/after number/after* aftersnum = code after the section number (see pseudo code below)</code>	
<code>¶/title/before title/before* aftersnum = code before the title (ie. after the section number box)</code>	
<code>¶/leader ¶/leaders = modifies the leader: the default is \cftdotfill{\cft¶dotsep}</code>	
<code>¶/dotsep = the space (in mu) between dots in the leader. Default is \cftdotsep and blank value</code>	
<code>is the same as \cftnodots</code>	
<code>¶/pagefont ¶/page/font ¶/pagenumber/font = modifies the font used to typeset the page number</code>	
<code>¶/page/after ¶/pagenumber/after afterpnum = macro expanded after the page number is typeset</code>	}

~~~~~ MEMO ~~~~~

numbered entries are typeset with this pseudo code:

```
{\cftXfont {\cftXpresnum SNUM\cftXaftersnum\hfil} \cftXaftersnumb TITLE}%
{\cftXleader}{\cftXpagefont PAGE}\cftXafterpnum\par
```

and for not numbered entries:

```
{\cftXfont TITLE}{\cftXleader}{\cftXpagefont PAGE}\cftXafterpnum\par
```

\lofsetup and \lotsetup behave the same.

\listofsetup{<list-name>}{<key=value>}

This command is provided to setup the parameters of user-defined lists created with tocloft's \newlistof and printed with \listofX (where X is the name given to the list). The keys are the same as ~\tocsetup, but there is no sub-keys for section, subsection *etc..*

8 Interface ↗ fancyhdr : \pagestyle

| | | |
|--|--|---|
| \pagestyle[<pagestyle-name>]{ | | |
| norule | | set \headrulewidth and \footrulewidth to 0pt |
| rule rule/width[* + -]=..., | rule/color=..., | set \headrulewidth and \footrulewidth to a given value |
| headrule=..., | rule/font=..., | To change the rule font/color |
| head/rule/width=..., | footrule=..., | redefines \headrule and \footrule (no value restores default) |
| head/rule/color=..., | foot/rule/width=..., foot/rule/skip=..., | dimensions |
| head/rule/font=..., | foot/rule/color=..., | for head and foot rules separately |
| head=..., | foot/rule/font=..., | <i>idem</i> |
| left=..., | foot=..., | header and footer (\fancyhead\fancyfoot) |
| head/left=..., | center=..., | right=..., by position |
| foot/left=..., | head/center=..., | head/right=..., for one side documents |
| head/left/even=..., | foot/center=..., | foot/right=..., <i>idem</i> |
| etc. | head/left/odd=..., | for two side documents |
| foot/right/even=..., | etc. | etc. |
| offset=..., | foot/right/odd=..., | changes \fancyhoffset |
| head/offset=..., | foot/offset=..., | |
| left/offset=..., | right/offset=..., | |
| head/left/even/offset=..., | head/left/odd/offset=..., | for two side documents |
| etc. | etc. | |
| foot/right/even/offset=..., | etc. | for two side documents |
| pagestyle=..., | | load the settings of another page style |
| \pagestyle* [<pagestyle-name>]{key = value} | | } |

\pagestyle resets all headers and footers, rules etc. and modifies the current (active) page style depending on the key=value pairs. \pagestyle[<page-style>] defines (or redefines) a page style, starting from default values. When a key is specified with no value, the default is used (f. ex. head/rule will be .4pt and foot/rule will be 0pt).

\pagestyle\* does not reset current values: keys that are not specified remains the same.

\pagestyle\* [<page-style>] defines or redefines <page-style>; the normal form starts from scratch, while the starred form \pagestyle\* keeps previous values if they are not overwritten, and if the <page-style> exists, of course.

foot/left/odd is the same as odd/foot/left etc. but twoside option is not checked... Even pages are ignored in oneside documents: this is LATEX, not fancyhdr nor **interfaces-fancyhdr**.

As usual, for dimensions or glues, +, - and \* keys are provided for addition, subtraction and scaling . And for code (font keys) the + and \* keys are provided: both append their content to the previous value.

9 Interface marks

`interface-marks` provides a few macros to make easier the use of ε -TeX \marks registers. ε -TeX defines 32 768 marks registers, making the use of them far more comfortable than “old” L<sup>A</sup>T<sub>E</sub>X \markboth, \markright, \leftmark and \rightmark. Who gets the correct – I mean *expected* – result with such commands ? Not me, I can say !

```
\marksthe{{named-mark}}{<content>}
\marksthecs{{named-mark}}{<cs-name>}
```

`\marksthe{{section}}{<content>}` will mark `<content>` into a mark register referenced by the control sequence `\marks@{section}`. **<content> is not expanded** (You can expand before `\marksthe` if you need it).

If the mark `\marks@{section}` is not defined, then a new mark register is allocated (with etex-\newmarks).

`\marksthecs{{subsubsection}}{<cs-name>}` will similarly mark the unexpanded content of `\cs-name` by means of the marks register `\marks@{subsubsection}`. `<cs-name>` is really the ***name of the control sequence***, not the control sequence itself. If `\cs-name` is undefined or `\relax`, nothing is marked, making `\marksthecs` suitable to use with `\lastsectiontitle etc.. at \sectionbreak` (see `interfaces-hypbmsec` p. 7).

```
\thefirstmarks{{named-mark}}
\thebotmarks{{named-mark}}
\thetopmarks{{named-mark}}
```

`\thefirstmarks{{chapter}}` expands to `\firstmarks\marks@{chapter}` (this is expandable), *i.e.* either the content of the first invocation of `\marksthe{{chapter}}` on the current page, if `\marks` was used on the current page, or the last invocation of `\marks` if no marks occurs on the current page.

`\thebotmarks{{chapter}}` expands to `\botmarks\marks@{chapter}` *i.e.* the content of the last invocation of `\marksthe{{chapter}}` (the most recent `\marks`).

`\thetopmarks{{chapter}}` expands to `\topmarks\marks@{chapter}` and this is the content of `\botmarks` at the time TeX shipped out the last page.

If there is no mark or if the mark register does not exist, expansion is null (*i.e.* nothing is done).

Those commands can be used in headers and footers, f.ex:

```
\sectionformat{section}
  mark=\marksthe{{section}}{\arabic{section}\hskip.4em\textrandom\hskip.4em#1}
\pagesetup{head/right=\thefirstmarks{{section}}}
```

```
\gettmarks\firstmarks|\botmarks|\topmarks{{named-mark}}{\control-sequence}
\getthefirstmarks{{named-mark}}{\control-sequence}
\getthebotmarks{{named-mark}}{\control-sequence}
\getthetopmarks{{named-mark}}{\control-sequence}
```

You may wish to retrieve the values of `\firstmarks`, `\botmarks` or `\topmarks` into a macro. This is the purpose of these commands: `\control-sequence` is defined as a parameterless macro whose replacement text is the content of the specified marks register.

If the marks register `\marks@{named-mark}` does not exist, the meaning of `\control-sequence` is undefined.

```
\ifmarksequal{\firstmarks}{\topmarks}{named-mark}{<true>}{<false>}
```

Pretty often, we want to compare the botmarks with firstmarks or the topmarks, to adapt the header and/or footer in case those marks are equal or different, *i.e.* in case the page contains a new section title or not:

`\ifmarksequal` expands conditionnally code `{<true>}` only if the specified marks register to compare are equal, otherwise `{<false>}`.

- `\firstmarks = \botmarks` \Leftrightarrow there is at most one section title on the current page;
- `\topmarks = \botmarks` \Leftrightarrow there is no section title on the current page;
- `\firstmarks = \topmarks` \Leftrightarrow the last section title continues on the current page.

Those assertions are true only if you mark the section title at `\sectionmark` and at `\sectionbreak`: this is easy when using `interfaces-hypbmsec`: `\sectionformat{subsection}{break=\marksthecs{{subsection}}{\lastsubsectiontoc}}`.

If the marks register `\marks@{named-mark}` does not exist, the `{<false>}` part is expanded.

10 Interface ↗ makecell : \repeatcell

`makecell` provides `\eline`, `\nline`, `\Xrows` to fill a table with empty cells, or repeated rows.

`interfaces` extends the possibility with `\repeatcell`. Although it is of general use, `\repeatcell` should be used softly in order to keep the `.tex` document intelligible (see the examples below).

While repeating cells, `\repeatcell` can also make calculations with:

- `\numexpr`,
- `\pgfmath` (\sim `\usetikz{pgfmath}` required in preamble)
- `$$\FPeval$$` (`\usepackage{fp}` required in preamble) \Rightarrow use `round` to display the results!

Inside formulas `\row` expands to the current row, `\column` to the current column number (`\row` and `\column` are 1-based and relative to `\repeatcell`).

`\countvalues{<comma separated list>}`

`\countvalues{<comma separated list>}` expands to the number of values in the list. This is also the number of steps a \LaTeX `\@for` loop would do on the list: in other words, **empty values are counted**. `\countvalues` is provided to allow you to adjust automatically the number of columns of a tabular, depending on your data, with the `*` operator: `\begin{tabular}{*{\countvalues{list}}{c}}`.

`\countvalue` is purely expandable in one expansion step. The `<comma separated list>` is expanded once, exactly like in \LaTeX `\@for` (*hint*: you can put `\@empty` before, to avoid expansion).

`\repeatcell[<alignment>]{<columns>}{<text>}{}{<rows>}{<cr|end>}` ← number of columns or comma separated list of values
`<text>` = the text to be inserted (repeated) in each cell
`` = the font inside the cell
`<rows>` = the number of rows (default is 1 row) or a comma separated list of value:
`{one, two, three, four, etc.}` Empty values **are not** ignored.
`<cr|end>` = the end-of-row sequence (default is `\backslash` but you may want `\crrcr\hline` or
`\tabularnewline...`)
end of row sequence is never inserted after the last row.
`text/row1 | text/row2 | text/row3 etc.` = If you want to change the text on the specified rows
`text/col1 | text/col2 | text/col3 etc.` = If you want to change the text on the specified columns
`font/row1 | font/row2 | font/row3 etc.` = If you want to change the font on the specified rows
`font/col1 | font/col2 | font/col3 etc.` = If you want to change the font on the specified columns
`cr/row1 | cr/row2 | cr/row3 etc.` = If you want to change the end-of-line sequence on the specified rows
`transpose` = Do you wish to invert rows and columns ? Then `transpose=true` ☺.
`any other key` = can be used to make calculations using `\numexpr` or `\pgfmathparse`
or `$$\FPeval$$`. Pay attention to put your formula inside curly brackets
`{$formula$}` if it contains a comma `,` or an equal `=` sign! }

- ☞ If `[<alignment>]` is specified, then each cell is placed into a `\makecell[<alignment>]` command.
- ☞ If both `text/row<i>` and `text/col<i>` are specified for the same cell, then `makecell \diaghead` is used to display the values in a diagonally divided cell.
- ☞ If you want to see (in the `.log` file) the code introduced into the tabular (or array) by `\repeatcell` type `\let\itf@message=\message` before `\repeatcell`.

Formulas: if you say: `x = $sin(10*\column$)` for example, then the formula is evaluated in each cell with `\pgfmath`. The result is stored into the control sequence `\x` and can be used in `text` or in any other following formula. The order in which the computed keys are given determines the order of evaluation... *of course!*

If you specify the number of columns (*resp.* of rows) as an integer, then `\repeatcell` works exactly as if you gave it a list of integers $\{1, 2, 3, 4, \dots, p\}$.

If you specify the rows as a comma separated list of values, the following sequences are also admissible in `text`:

- | | |
|------------------------------------|---|
| <code>\R\row</code> | refer to the value of the list for the current row. <code>\r</code> is the same (unless you defined a formula named <code>r</code>). |
| <code>\R{\langle n \rangle}</code> | to the value of the list for row $\{\langle n \rangle\}$ |

If the number of rows is specified as an integer, then `\R\row`, `\r` and `\row` are equal (to the current row index).

Similarly if you specify the columns as a comma separated list of values, the following sequences are provided:

- `\C\column` refer to the value of the list for the current column. `\c` is the same, unless you defined a formula named `c`.
- `\C{\<m>}` to the value of the list for column `{<m>}`

If the number of columns is specified as an integer, then `\C\row`, `\c` and `\col` are equal (to the current column index).

If the values in the lists are numeric, `\R`, `\C`, `\r` and `\c` can be used in formulas as well, *of course...*

Important note: The mechanism of `\halign` does not allow to insert tab alignment characters into the arguments of `\repeatcell`. Therefore, to be able to define a `text` key containing a tabular or array, the `&` alignment `tab` character is activated (`catcode 13`) at the beginning of `\repeatcell`. This activation has no effect if `\repeatcell` is in the replacement text of one of your macro, because the category code of `&` is fixed at the time your macro was read. Thus, use the control sequence `\tab` instead of `&` in this case. `\tab` is only defined during the time `\repeatcell` does its job.

~~~~~ A few examples are better than a 10 pages description ! ~~~~~

## 10.1 † Conversion to scaled points

```
\def\listofunits{ pt , cm , mm , bp , dd , pc , in , cc ,sp }
\begin{tabular}{|>{\bfseries}c|*{9}{|r}|} \hline
\thead{units}
& \repeatcell[c]\listofunits{ text = \c, font=\bfseries } \\ \hline
\thead{value in\\ scaled\\ points}
& \repeatcell\listofunits{%
    sp =\number\dimexpr 1\c\relax,
    text=\numprint[sp]\sp} \\ \hline
\end{tabular}
```

| units                        | pt        | cm           | mm         | bp        | dd        | pc         | in           | cc         | sp   |
|------------------------------|-----------|--------------|------------|-----------|-----------|------------|--------------|------------|------|
| value in<br>scaled<br>points | 65 536 sp | 1 864 679 sp | 186 467 sp | 65 781 sp | 70 124 sp | 786 432 sp | 4 736 286 sp | 841 489 sp | 1 sp |

## 10.2 † Variable length cell

```
\newcommand\FAMILY[3]{%
\begin{tabular}{|*{3}{>{\centering\arraybackslash}\m{.12\linewidth}}|} \hline
Mother & Father & Children \\ \hline
#1 & #2 & \makecell{\repeatcell[#3]{text=\c, transpose}} \\ \hline
\end{tabular}}
\FAMILY{Johanne}{Georg}{Priscilla, John, Tom}
\FAMILY{Nut}{Geb}{Isis, Osiris, Seth, Nephthys}
```

| Mother  | Father | Children                           |
|---------|--------|------------------------------------|
| Johanne | Georg  | Priscilla<br>John<br>Tom           |
| Nout    | Geb    | Isis<br>Osiris<br>Seth<br>Nephthys |

## 10.3 † Empty rows

```
\begin{tabular}{*{6}{|c|}} \hline
\repeatcell{6}{ text/col1=\row, rows=4, end=\hline } \\ \hline
\end{tabular}
\begin{tabular}{*{4}{|c|}} \hline
\repeatcell{6}{ text/col1=\row, rows=4, end=\hline , transpose} \\ \hline
\end{tabular}
```

|   |  |  |  |
|---|--|--|--|
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |

|   |   |   |   |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
|   |   |   |   |
|   |   |   |   |
|   |   |   |   |
|   |   |   |   |

## 10.4 † color palette

```
\begin{tabular}{*{4}c}
\repeatcell{8}{transpose=true, rows=4,
red      = {$\$round(\c/8*255,0)$$},
greeni   = \red,           bluei    = \red,
greenii  = {$\$round(255-\red,0)$$}, blueii   = \red,
greeniii = \red,          blueiii  = {$\$round(255-\red,0)$$},
greeniv  = \greenii,       blueiv   = \blueii,
text/row1=\cellcolor[RGB]{\red,\greeni,\bluei}(\red,\greeni,\bluei),
text/row2=\cellcolor[RGB]{\red,\greenii,\blueii}(\red,\greenii,\blueii),
text/row3=\cellcolor[RGB]{\red,\greeniii,\blueiii}(\red,\greeniii,\blueiii),
text/row4=\cellcolor[RGB]{\red,\greeniv,\blueiv}(\red,\greeniv,\blueiv)}
\end{tabular}
```

|               |               |               |               |
|---------------|---------------|---------------|---------------|
| (32,32,32)    | (32,223,32)   | (32,32,223)   | (32,223,223)  |
| (64,64,64)    | (64,191,64)   | (64,64,191)   | (64,191,191)  |
| (96,96,96)    | (96,159,96)   | (96,96,159)   | (96,159,159)  |
| (128,128,128) | (128,127,128) | (128,128,127) | (128,127,127) |
| (159,159,159) | (159,96,159)  | (159,159,96)  | (159,96,96)   |
| (191,191,191) | (191,64,191)  | (191,191,64)  | (191,64,64)   |
| (223,223,223) | (223,32,223)  | (223,223,32)  | (223,32,32)   |
| (255,255,255) | (255,0,255)   | (255,255,0)   | (255,0,0)     |

*Note:* `makecell` interact very badly with `colortbl`. Therefore, with `\cellcolor` or `\rowcolor`, the optional `[⟨alignment⟩]` parameter of `\repeatcell` does not give good results... `colortbl` remain a rather rough package, and deserve a finer implementation... in future.

## 10.5 † tabularcalc-like tables (with dcolumn package) and pgfmath

```
\def\listoffx{ -4 , -2 , 0 , 2.25 , 7 }
$\$ \begin{tabu}{|c<{\btopstrut}|*{5}{|D..{5.5}|}|} \hline
\rowfont{\large\color{blue}}
x &\repeatcell{\listoffx{ text=\C\column }} &&&& \\ \hline
f(x)=2x-3 &\repeatcell{\listoffx{ fx = \$2*\c-3\$, text=\fx }} &&&& \\ \hline
x^2 &\repeatcell{\listoffx{ xsquare = \$\C\column^2\$, text = \xsquare }} &&&& \\ \hline
h(x) = \sqrt{x^2+1} &\repeatcell{\listoffx{ x=\C\column, hx = {\$sqrt(pow(\x,2)+1)\$}, text = \hx }} &&&& \\ \hline
\end{tabu} $$
```

| x                       | -4      | -2      | 0       | 2.25    | 7       |
|-------------------------|---------|---------|---------|---------|---------|
| $f(x) = 2x - 3$         | -11.0   | -7.0    | -3.0    | 1.5     | 11.0    |
| $x^2$                   | -16.0   | -4.0    | 0.0     | 5.0625  | 49.0    |
| $h(x) = \sqrt{x^2 + 1}$ | 4.12310 | 2.23606 | 1.00000 | 2.46221 | 7.07106 |

\$\pgfmath\$ computations

| x                       | -4        | -2       | 0        | 2.25    | 7        |
|-------------------------|-----------|----------|----------|---------|----------|
| $f(x) = 2x - 3$         | -11.00000 | -7.00000 | -3.00000 | 1.50000 | 11.00000 |
| $x^2$                   | 16.00000  | 4.00000  | 0.00000  | 5.06250 | 49.00000 |
| $h(x) = \sqrt{x^2 + 1}$ | 4.12311   | 2.23607  | 1.00000  | 2.46221 | 7.07107  |

And the same with `FPeval` (fp package)

## 10.6 Double entry array: the Body Mass Index

| height in cm | 190  | 180  | 170  | 160  | 150  | 140  | 130  |                 |
|--------------|------|------|------|------|------|------|------|-----------------|
| weight in kg |      |      |      |      |      |      |      |                 |
| 30           | 8.3  | 9.3  | 10.4 | 11.7 | 13.3 | 15.3 | 17.8 | denutrition     |
| 45           | 12.5 | 13.9 | 15.6 | 17.6 | 20.0 | 23.0 | 26.6 | skinniness      |
| 50           | 13.9 | 15.4 | 17.3 | 19.5 | 22.2 | 25.5 | 29.6 | normal          |
| 55           | 15.2 | 17.0 | 19.0 | 21.5 | 24.4 | 28.1 | 32.5 | obesity         |
| 60           | 16.6 | 18.5 | 20.8 | 23.4 | 26.7 | 30.6 | 35.5 | serious obesity |
| 65           | 18.0 | 20.1 | 22.5 | 25.4 | 28.9 | 33.2 | 38.5 | morbid obesity  |
| 70           | 19.4 | 21.6 | 24.2 | 27.3 | 31.1 | 35.7 | 41.4 |                 |
| 80           | 22.2 | 24.7 | 27.7 | 31.3 | 35.6 | 40.8 | 47.3 |                 |
| 90           | 24.9 | 27.8 | 31.1 | 35.2 | 40.0 | 45.9 | 53.3 |                 |
| 95           | 26.3 | 29.3 | 32.9 | 37.1 | 42.2 | 48.5 | 56.2 |                 |
| 100          | 27.7 | 30.9 | 34.6 | 39.1 | 44.4 | 51.0 | 59.2 |                 |
| 110          | 30.5 | 34.0 | 38.1 | 43.0 | 48.9 | 56.1 | 65.1 |                 |
| 120          | 33.2 | 37.0 | 41.5 | 46.9 | 53.3 | 61.2 | 71.0 |                 |

| height in cm \ weight in kg | 30   | 45   | 50   | 55   | 60   | 65   | 70   | 80   | 90   | 95   | 100  | 110  | 120  |
|-----------------------------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| height in cm                |      |      |      |      |      |      |      |      |      |      |      |      |      |
| 190                         | 8.3  | 12.5 | 13.9 | 15.2 | 16.6 | 18.0 | 19.4 | 22.2 | 24.9 | 26.3 | 27.7 | 30.5 | 33.2 |
| 180                         | 9.3  | 13.9 | 15.4 | 17.0 | 18.5 | 20.1 | 21.6 | 24.7 | 27.8 | 29.3 | 30.9 | 34.0 | 37.0 |
| 170                         | 10.4 | 15.6 | 17.3 | 19.0 | 20.8 | 22.5 | 24.2 | 27.7 | 31.1 | 32.9 | 34.6 | 38.1 | 41.5 |
| 160                         | 11.7 | 17.6 | 19.5 | 21.5 | 23.4 | 25.4 | 27.3 | 31.3 | 35.2 | 37.1 | 39.1 | 43.0 | 46.9 |
| 150                         | 13.3 | 20.0 | 22.2 | 24.4 | 26.7 | 28.9 | 31.1 | 35.6 | 40.0 | 42.2 | 44.4 | 48.9 | 53.3 |
| 140                         | 15.3 | 23.0 | 25.5 | 28.1 | 30.6 | 33.2 | 35.7 | 40.8 | 45.9 | 48.5 | 51.0 | 56.1 | 61.2 |
| 130                         | 17.8 | 26.6 | 29.6 | 32.5 | 35.5 | 38.5 | 41.4 | 47.3 | 53.3 | 56.2 | 59.2 | 65.1 | 71.0 |

The same with transpose

**Note that fp computation can be time consuming...** depending on the formula and the number of cells to fill in. While preparing first such a table, give to the computed keys a constant value, and plug the formulas only when you are satisfied with the displayed tabular...

## 11 Interface ☞ enumitem : \setlist

```
\setlist [⟨environment, level⟩]{
    label = the label to be used
    font = the label font
    topsep = vertical space before the list
    partopsep = vertical space
    parsep = value of \parskip inside the list
    itemsep = vertical space between two items
    leftmargin = the list left margin
    rightmargin = the list right margin
    listparindent = the value of \parindent inside the list
    labelwidth = the width of the label box
    labelsep = horizontal space between label and following text
    itemindent = the indentation of the items
\setlist* [⟨environment, level⟩]{key=value}
```

Enumerated lists, description lists or itemize lists can be defined by `enumitem` with the command `\newlist`.

For example: `\newlist{mylegal}{enumerate}{10}`

The list can be customized with `\setlist`, for example:

```
\setlist[mylegal,1]{label*=\arabic*.}
\setlist[mylegal,2]{label*=\arabic*.}
\setlist[mylegal,3]{label*=\roman*)}
```

With `interfaces-enumitem` you may specify more than one level at a time in `\setlist`, (and even more than one environment at a time):

```
\setlist[mylegal,1,2]{label*=\arabic*.}
\setlist[mylegal,3]{label*=\roman*)}
```

A list can also be **modified** with `\setlist*`: keys that are not specified remain the same.

```
\setlist*[mylegal,1,2,3]{align=left, leftmargin=*}
```

or

```
\setlist*[mylegal]{align=left, leftmargin=*}
```

### `\interitem`

Just like  $\mathcal{M}\mathcal{S}$  `\intertext` in math formulæ, `\interitem` allows to put some non intended text inside a list environment. Example:

```
\begin{enumerate}[label=\roman*)], ,parsep=0pt]
\item First item in the enumerate envir
\interitem or (\intertext)
\item Second item
\end{enumerate}
```

The example give in real world:

- i) First item in the enumerate envir
- or: (\intertext)
- ii) Second item

And the following follows...

## 12 Interface ⇔ truncate

The package `truncate` by Donal Arseneau provides the command:

```
\truncate[⟨TruncateMarker⟩]{⟨width⟩}{⟨text⟩}
```

to trunc the given `⟨text⟩` to the specified `⟨width⟩`, by means of `⟨TruncateMarker⟩` if the the truncation occurs.

```
\trunctext[⟨TruncateMarker⟩]{⟨width⟩}{⟨text⟩}
```

`\trunctext` is the same as `\truncate` but it leaves `{text}` to its natural width if it has not to be truncated (*i.e.* if  $\langle width \rangle \leq \text{width of } \langle text \rangle$ ). Example:

```
truncate: \truncate{<5cm>}{Hello world!}
trunctext: \trunctext{<5cm>}{Hello world!}
```

Hello word!

Hello world!

This corresponds to the `[fit]` option of the `truncate` package.

```
\htruncadjust{⟨total width⟩}[⟨x1, x2, ..., xn⟩]{⟨text1, text2, ..., textn⟩}
```

This macro will measure the width of `text1,text2,...,textn` and adjust them for truncation:

| text              | width | adjusted width                                                                          | $\gamma$ (truncated?)                                                                    | Returned width                                                              |
|-------------------|-------|-----------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------|
| text ₁ | $w_1$ | $w'_1 = \min\left(w_1, \frac{x_1}{\sum x_i} \times \langle total\ width \rangle\right)$ | $\gamma_1 = \begin{cases} 1 & \text{if } w'_1 < w_1 \\ 0 & \text{otherwise} \end{cases}$ | $\omega_1 = w'_1 + \gamma_1 \cdot \Phi \cdot \frac{x_1}{\sum \gamma_i x_i}$ |
| text ₂ | $w_2$ | $w'_2 = \min\left(w_2, \frac{x_2}{\sum x_i} \times \langle total\ width \rangle\right)$ | $\gamma_2 = \begin{cases} 1 & \text{if } w'_2 < w_2 \\ 0 & \text{otherwise} \end{cases}$ | $\omega_2 = w'_2 + \gamma_2 \cdot \Phi \cdot \frac{x_2}{\sum \gamma_i x_i}$ |
| text ₃ | $w_3$ | $w'_3 = \min\left(w_3, \frac{x_3}{\sum x_i} \times \langle total\ width \rangle\right)$ | $\gamma_3 = \begin{cases} 1 & \text{if } w'_3 < w_3 \\ 0 & \text{otherwise} \end{cases}$ | $\omega_3 = w'_3 + \gamma_3 \cdot \Phi \cdot \frac{x_3}{\sum \gamma_i x_i}$ |
| text ₄ | $w_4$ | $w'_4 = \min\left(w_4, \frac{x_4}{\sum x_i} \times \langle total\ width \rangle\right)$ | $\gamma_4 = \begin{cases} 1 & \text{if } w'_4 < w_4 \\ 0 & \text{otherwise} \end{cases}$ | $\omega_4 = w'_4 + \gamma_4 \cdot \Phi \cdot \frac{x_4}{\sum \gamma_i x_i}$ |
|                   |       | $\Phi = \text{freespace} = \langle total\ width \rangle - \sum w'_i$                    |                                                                                          | $\sum \omega_i \leq \langle total\ width \rangle$                           |

Another example of a tabular made with  $\sim\backslash\text{repeatcell}$

The widths can be accessed with `\htruncwidth{⟨i⟩}` and then:

```
\trunctext[⟨TruncateMarker⟩]{\htruncwidth{⟨i⟩}}{⟨texti⟩}.
```

$x_i$  can be any real number provided that  $\sum_i x_i \leq 16\,383 = \text{\maxdimen}$  in points.

Measuring the widths occur inside a `\hbox`, thus `text1,...,textn` can contain macros, non expandable material *etc..*. For example:

`\htruncadjust{4cm}[2,1]{Hello,world}` will prepare the widths so that their sum does not exceed 4 centimeters, and  $w_1 = 2 \times w_2$ .

`\htruncadjust{4cm}[2,1,2]{The,World,Is flat}` will prepare the widths so that their sum does not exceed 4 centimeters, and  $w_1 = w_3 = 2 \times w_2$ .

If the optional parameters  $x_i$  are not given, then they are equal to the natural width of `{text1,...,textn}`. In other words, `text1,...,textn` are truncated to a width proportional to their natural widths. (And the sum of their widths will not exceed `⟨total width⟩`).

This macro has been designed to cleverly truncate the text of headers, when they contain titles on both sides... Usually then, there are only two pieces of text whose width have to be adjusted for truncation...

```
\htruncadjust{3cm}{first quite long,second short,third} % natural widths
\trunctext{\htruncwidth{1}}{first quite long}
\trunctext{\htruncwidth{2}}{second short}
\trunctext{\htruncwidth{3}}{third}
```

→ first quite...second...third (39.05698pt, 33.65436pt, 12.64691pt, total=85.35825pt).

```
\htruncadjust{3cm}[3,2,2]{first quite long,second short,third} % first is 3/2 of the two others
```

→ first quite...secon...third (41.01503pt, 27.34372pt, 17.00078pt, total=85.35953pt).

## 13 Interface ↗ appendix and the appendices environment

appendix provides a way to write appendices inside an environment. This feature is suitable for short appendices like the one included in this document. `interfaces-appendix` gives an easier control on some parameter for the appendix with standard (optional) key names.

---

```
\begin{appendices}[%  
    section = section | chapter,      alternatively, the key chapter is provided  
    title = the appendix title,       if left blank, title is removed: [page] option of appendix  
    intoc = true | false,            put an entry in t.o.c. ?  
    intoc/title = the entry in t.o.c.] if blank, no title in t.o.c., if no value: the appendix title  
    appendix content ...  
\end{appendices}
```

---

## 14 Interface ↗ embedfile : \embedthisfile

```
\embedthisfile  
\embedthisfile*
```

`\embedthisfile` will embed into the final .pdf file, the current .tex file and all .tex files that are included (with `\include`) in it. If `\includeonly` is used, the only files embeded (*i.e.* attached) to the .pdf file are the ones specified by `\includeonly`.

`\embedthisfile*` embeds only the main file.

This macro could disappear from `interface-embedfile` if it is added to the main package... Depending on Mr Oberdiek moods ☺.

## 15 Interface ↗ environ

`environ` defines `\Renewenviron` but I found out it was usefull to have `\ReNewEnviron` (note the case!).

Therefore: `\let\ReNewEnviron\RenewEnviron` is made by `interfaces` (`\AfterPackage{environ}`).

## 16 Interface umrand

`umrand` defines the macro `\ifnotempty` which is already defined with a different meaning in `etextools`. `interfaces` provides a turn around to avoid redefinition.

For this turn around, `interfaces` must be loaded before `umrand` (because we use `\BeforePackage{umrand}`). This is – really – the only exception!

## 17 Interface pgfkeys (↗ pgfmanual)

pgfkeys's the base of all... wonderfull and inimitable! ☺

The following handlers allows to build fairly quickly a key=value interface for commands in almost any case:

### ~~~~~ SUMMARY ~~~~~

|                                            |                                                                                                                                          |
|--------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| <code>./blank</code>                       | replace blank value by another value                                                                                                     |
| <code>/is private</code>                   | throws an error if the key is used in an assignment                                                                                      |
| <code>/store in cs .estore in cs</code>    | like <code>/store in</code> but with <code>\csname...\endcsname</code>                                                                   |
| <code>/true /false</code>                  | <code>/is choice</code> shortcut for boolean keys                                                                                        |
| <code>/on /off</code>                      |                                                                                                                                          |
| <code>/is multiple choice</code>           | like <code>/is choice</code> but more than one choice can be selected at a time                                                          |
| <code>/is enum</code>                      | <code>/is choice</code> shortcut for enumerated keys                                                                                     |
|                                            | also provides the <code>/key/.number</code> key to get the numeric index of the selected choice.                                         |
| <code>/store as code</code>                | store value ( <i>expanded value</i> ) in the key. Created <code>+</code> and <code>*</code> keys to append code                          |
| <code>/estore as code</code>               |                                                                                                                                          |
| <code>/store as code in cs</code>          | store value ( <i>expanded value</i> ) in a cs-name rather than in the key. Created <code>+</code> and <code>*</code> keys to append code |
| <code>/estore as code in cs</code>         |                                                                                                                                          |
| <code>/store as color</code>               | store the color ( <i>the color command</i> ) in the key.                                                                                 |
| <code>/store as color command</code>       |                                                                                                                                          |
| <code>/store as color in cs</code>         | the same, store in the cs-name rather than in the key                                                                                    |
| <code>/store as color command in cs</code> |                                                                                                                                          |
| <code>/store as number</code>              | store an integer in the key (- + and * keys are provided to add, subtract and multiply)                                                  |
| <code>/estore as number</code>             |                                                                                                                                          |
| <code>/store as number in cs</code>        | the same, but stores into a cs-name rather than in the key                                                                               |
| <code>/estore as number in cs</code>       |                                                                                                                                          |
| <code>/store as glue</code>                | store a glue in the key (+ - and * keys are provided to add, subtract or scale)                                                          |
| <code>/estore as glue</code>               |                                                                                                                                          |
| <code>/store as glue in cs</code>          | the same, store in a given named control sequence                                                                                        |
| <code>/estore as glue in cs</code>         |                                                                                                                                          |
| <code>/store in skip cs</code>             | store the glue inside a <code>\skip</code> register rather than in a macro                                                               |
| <code>/store as dimen</code>               | store a dimen in the key (with + - and * keys for addition, subtraction and scaling)                                                     |
| <code>/estore as dimen</code>              |                                                                                                                                          |
| <code>/store as dimen in cs</code>         | the same, store in a given named control sequence rather than in the key itself                                                          |
| <code>/estore as dimen in cs</code>        |                                                                                                                                          |
| <code>/store in dimen cs</code>            | store the dimension into a <code>\dimen</code> register rather than in a macro                                                           |

The command `\pgfkeysmeaning` is provided (with `\providetcommand`): it expands to the meaning of the given pgfkey.

### 17.1 † Handler definition: `./blank`

A new handler `./blank` is defined to allow automatic replacement of empty or blank values by the “default” value specified:

```
\pgfkeys{/base/path/my key/.code=The key #1,%  
        /base/path/my key/.default=has its default value!,%  
        /base/path/my key/.blank=was left blank!}  
  
\pgfkeys{/base/path/my key,           produces The key has its default value!  
        /base/path/my key=}            produces The key was left blank!
```

The following make blank value behave the same as default :

```
\pgfkeys{/base/path/my key/.blank=\pgfkeysnovalue}
```

And If you wish to forbid blank values you can add:

```
\pgfkeys{/base/path/my key/.value required}
```

With `interface-pgfkeys` it is also possible to give a `./default` or/and a `./blank` value to a handler, or on the other hand to say that a handler must have `(./value required)` or does not suffer `(./value forbidden)` a value. (this feature was not implemented in `pgfkeys.code.tex`).

## 17.2 † Handler definition: `./is private`

This handler is mainly addressed to package writers. A private key does not suffer any assignment: this is a package constant:

```
\pgfkeys{/base/path/@my key/.is private=mypackage}
```

Then any assignment to @my key will throw an error, explaining that @my key is private to the package mypackage.

Similarly, there is a `./is private to class` handler (for `\ClassError` instead of `\PackageError`).

## 17.3 † Handler definition: `./store in cs` and `./estore in cs`

A new handler `./store in cs` is defined to store the value of a key in a named control sequence:

```
\pgfkeys{%
    /base/path/my key/.store in cs=some word,% will define : \"some word"
}
```

## 17.4 † Handlers definition:: `./true`, `./false`, `./on` and `./off`

A key of type `./true` or `./false` can take the only values `true` or `false` and its default value is `true` (unless `./default=false` is explicitly given to `\pgfkeys`).

Similarly, a key of type `./on` or `./off` can take the only values `on` or `off` and its default value is `on` (unless `./default=off` is explicitly given to `\pgfkeys`).

These are shortcuts and a special case of `./is choice` keys:

```
\pgfkeys{
    my key/.true = code if the value is true,
    my key/.false = code if the value is false,
    my key2/.on = code if the value is on,
    my key2/.off = code if the value is off }
```

## 17.5 † Handler definition: `./is multiple choice`

A key defined with `./is multiple choice` behaves like a `./is choice` key but more than one choice can be selected at a time:

```
\pgfkeys{/path/.cd,
    format/.is multiple choice,
    format/bold/.code= ...,
    format/italic/.code= ...,
    format/color/.code= ... }
```

Possible values for key `format` are restricted to subkeys: `bold`, `italic` and `color`. But unlike simple `./is choice` keys, it is possible to use:

```
\pgfkeys{/path/format={bold,italic,color=...}}
```

this is the same as:

```
\pgfkeys{/path/format=bold,/path/format=italic,path/format={color=...}}
```

*The bookmark key of ↳ \sectionformat (in interface-titlesec) is a good example.*

## 17.6 † Handler definition: `./is enum`

A key defined with `./is enum` behaves like a `./is choice` key: its possible values are limited to sub-keys. The name of the selected sub-key is stored into the key.

```
\pgfkeys{/path/.cd,
  enumKey./is enum={one,two,three}
}
```

`enumKey` can be `one` or `two` or `three` and nothing else.

```
\pgfkeys{ enumKey=two }
```

assigns `two` to the value of the key: `\pgfkeysvalueof{enumKey}` is `two`.

`\pgfkeysvalueof{enumKey./number}` is **1** (zero based index)

## 17.7 † Handlers definition: `./store as code`, `./estore as code`

This handler stores the value given to the key in the key itself. Furthermore, a starred key `*` and a `+` keys are defined such that they append their value to the stored key.

```
\pgfkeys{/test./store as code,
  /test=some code,
  /test+= the following follows... }
```

Defines the key `/test` so that `\pgfkeysvalueof{/test}` expands to `some code`. `/test+` appends “the following follows” to the previous value of the key. (`/test*` does the same).

If you prefer not to have star keys `*`, or `+` keys, or only one of these, just say, for example:

```
\pgfkeys{/test./store as code=+}
```

*This feature is very important for `interfaces`: almost any `font` keys are defined with the handler `./store as code`.*

*A note for programmers:* `./store as code` defines three keys: `test`, `test+` and `test*`. (The two former are the same). The “append code” keys (`+` and `*`) call the *main key* `test`. This is handy because if you append code to the *main key* (with `pgfkeys ./add code` or `./append code`) then this code will be expanded also in case the `+` or `*` keys are called.

There is also a `./estore as code` handler which expands the given code before storing it in the key.

## 17.8 † Handlers definition: `./store as code in cs=<cs-name>`, `./estore as code in cs=<cs-name>`

This handler works like `./store as code`: the content is stored in the `cs-name` given as value for the handler, but not inside the key itself (useful for `key=value` interfaces to other packages).

The `+` and `*` keys are provided: both append their value to the content of the given control sequence.

Typically, `/font` keys are defined this way for interface with other packages.

There is also a `./estore as code in cs` handler which expands the given code before storing it in the `<cs-name>`.

## 17.9 † Handler definition: `./store as color`

When specifying a color, did you say:

|                                   |                                 |
|-----------------------------------|---------------------------------|
| <code>color=\color{yellow}</code> | <code>color=\normalcolor</code> |
| <code>color=blue</code>           | <code>color=[rgb]{1,0,0}</code> |

In any case, `./store as color` will transform the value as follow:

|                           |                                                                              |
|---------------------------|------------------------------------------------------------------------------|
| <code>color=yellow</code> | <code>color=default</code> defined by <code>interfaces</code> if not already |
| <code>color=blue</code>   | <code>color=[rgb]{1,0,0}</code> defined <code>\AtBeginDocument</code>        |

In any case, `./store as color command` will transform the value as follow:

|                                 |                                       |
|---------------------------------|---------------------------------------|
| <code>color=\normalcolor</code> | <code>color=\color{yellow}</code>     |
| <code>color=\color{blue}</code> | <code>color=\color[rgb]{1,0,0}</code> |

Please! Dont write: `\def\myBlue{blue}`    `color=\myBlue`.

Use `\colorlet{myBlue}{blue}` instead, or `\definecolor`.

As for code, the color is stored inside the key.

**17.10 † Handler definition: `./store as color in cs=<cs-name>`**

The same as `./store as color`, but the color is stored in the given control sequence, rather than in the key itself.

There is also a `./store as color command in cs` handler. It is used to set the value to `\columnseprulecolor` by the key `columns/rule/color` of `\tocsetup`.

**17.11 † Handlers definition: `./store as number, ./estore as number`****17.12 † Handlers definition: `./store as number in cs, ./estore as number in cs`****17.13 † Handler definition: `./store as glue`**

This is quite the same as `./store as code`, but the given code is surrounded by `\glueexpr...``\relax...` and is stored inside the key.

Furthermore:

the `+` key adds to the length: `\glueexpr old value + < argument > \relax`.

the `-` key subtracts: `\glueexpr old value - < argument > \relax`

the `*` key scales: `\glueexpr old value * < argument > \relax`

The `<argument>` is **not expanded** but the content of the key is ready to be used in place of a length. Scaling of two forms are allowed: *fractionnal* and *real*. *Real* scaling is converted to a fraction of 65536:

```
\pgfkeys{/test/.store as glue,
          /test=12pt plus5pt,
          /test*=.5,
          /test+=\parskip,
          /test*=4/6
}
\pgfkeysmeaning{/test}
→ macro:->\glueexpr \glueexpr \glueexpr \glueexpr 12pt plus5pt\relax *32768/65536\relax +\glueexpr \parskip\relax \relax *4/6\relax
```

The value of `\parskip` is not expanded at `\pgfkeys` time, but only when the value of the key is used (for an assignment...)

There is also a handler called `./estore as glue` if you really want the value to be expanded (with `\the`) at the time of `\pgfkeys` setting.

**17.14 † New handler: `./store as dimen`**

Just like `./store as glue` but with `\dimexpr` instead of `\glueexpr`.

There is also a `./estore as dimen` handler, for immediate expansion.

**17.15 † New handler: `./store as glue in cs=<cs-name>`**

This is like `./store as glue` but the the value is assigned to a macro whose name is given, rather than in the key itself. Expansion *does not* occur `\pgfkeys` time: the macro contains the glue definition in `\glueexpr...``\relax` expressions.

`+` (to add) `-` (to subtract) and `*` (to scale) keys are provided.

There is also a `./estore as glue in cs` handler, and expansion occurs when the key is set.

## 17.16 † New handler: `./store as dimen in cs=<cs-name>`

Just like `./store as glue in cs` but with `\dimexpr` instead of `\glueexpr`.

## 17.17 † New handler: `./estore as dimen in cs=<cs-name>`

This handler as a particularity: if the value is given without dimension, `pt` is assumed (that is... points).

This is the way the key `size` is defined for the command  $\sim\backslash\text{changefont}$  (in `interfaces-LaTeX`). `size` defines `\f@size`, the macro used by L^AT_EX to store the font size (and `size+` adds, `size-` subtracts, `size*` scales). But `\f@size` is stored in points, without the mention of the dimension: `pt`.

With its default units in `pt`, `./estore as dimen in cs` is the handler of the situation:

## 18 Interface ↗ etoolbox - etoolbox-like commands

May be they will be added to `etoolbox.sty` one day or another... and disappear from `interfaces` ...

### 18.1 † New macros provided with `\providetcommand`

|                                               |                                                    |
|-----------------------------------------------|----------------------------------------------------|
| <code>\removefrom {hook}{&lt;code&gt;}</code> | <code>\removefromstart{hook}{&lt;code&gt;}</code>  |
| <code>\gremovefrom{hook}{&lt;code&gt;}</code> | <code>\gremovefromstart{hook}{&lt;code&gt;}</code> |
| <code>\eremovefrom{hook}{&lt;code&gt;}</code> | <code>\eremovefromstart{hook}{&lt;code&gt;}</code> |
| <code>\xremovefrom{hook}{&lt;code&gt;}</code> | <code>\xremovefromstart{hook}{&lt;code&gt;}</code> |

`\removefrom` removes the first occurrence of `{<code>}` from a `<hook>`. The command is robust.

`{<code>}` should not contain groups: `{...}` but parameter characters are allowed, which need not to be doubled. The match for `{<code>}` inside the replacement text of `<hook>` is category-code sensitiv.

If the first occurrence of `{<code>}` into `<hook>` has been removed

then `\ifin@` is set to `\iftrue`

otherwise `\ifin@` is set to `\iffalse` (`<hook>` does not contain `{<code>}`).

this allows loops of the form:

`\in@true`

`\@whilesw\ifin@\fi{\removefrom{hook}{<code>}}`

☞ `\eremovefrom` and `\xremovefrom` expand `{<code>}` before trying to remove it from `<hook>`. Only `{<code>}` is expanded, the current replacement text of `<hook>` is not.

☞ `g` and `x` versions make the assignment to `<hook>` global.

☞ `\removefromstart` remove `{<code>}` only if it appears at the very beginning of `<hook>`.

## 19 Interface ↗ scrfile

### 19.1 † Star forms for `\BeforeFile` and `\AfterFile`

`scrfile` provides `\BeforeFile` and `\AfterFile` in order to execute some code when a file is loaded. The hooks are automatically reset to `\relax` after the file has been loaded, in other words, they are expanded at most one only time. This can be avoided by using `\BeforeFile*` and `\AfterFile*`, for example:

```
\BeforeFile*{\jobname.toc}{\begin{multicols}{2}}
\AfterFile*{\jobname.toc}{\end{multicols}}
```

The hooks *before* and *after* will be executed **every time** `\jobname.toc` is loaded.

This feature is used in `\tocsetup{twocolumns}`.

## 19.2 † Allowing \BeforeFile and \AfterFile when \@input is invoked

The hooks \BeforeFile and \AfterFile are executed only when the file is loaded with \InputIfFileExists: indeed this macro is redefined by `scrfile`.

`interfaces` redefine \@input in order to execute those hooks also in case \@input is used: this is useful because .toc files are input with \@input rather than with \InputIfFileExists.

## 20 List of corrected or patched commands

Here is the complete list of patched commands in other packages.

The modification occurs only if the given module of `interfaces` is loaded. For example, \addcontentsline is modified only if packages `titlesec` and `bookmark` are loaded, *and if* `interface-titlesec` and `interface-bookmark` are loaded.

### 20.1 † L^AT_EX

|   | Patched command                                                          | Type | Importance                          | Module           | ref. page |
|---|--------------------------------------------------------------------------|------|-------------------------------------|------------------|-----------|
| ❶ | \@pkgextension<br>\@clsextension<br>\@ifpackageloaded<br>\@ifclassloaded |      | minor<br>optimisation or correction | interfaces-LaTeX | 71        |

Those commands are replaced by their equivalent in the package `ltxcmds`:

```
\ltx@pkgextension \ltx@clsextension \ltx@ifpackageloaded \ltx@ifclassloaded
```

This is because L^AT_EX defines those commands \onlypreamble, while the equivalent macros in `ltxcmds` can be used inside `document` as well.

|   | Patched command  | Type    | Importance                      | Module                                   | ref. page |
|---|------------------|---------|---------------------------------|------------------------------------------|-----------|
| ❷ | \addcontentsline | prepend | important<br>but no side effect | interface-titlesec<br>interface-bookmark | 39        |

\addcontentsline does a lot a thing and in particular it expands the \bookmark command (if the `bookmark` package is loaded, otherwise it's `hyperref`). As long as `\sectionformat` can drive the attributes of the bookmark for each section, it is necessary to expand \bookmarksetup with the right options before the section-bookmark is written.

This is achived by \itf@ttl@BKM, placed at the beginning of \addcontentsline.

|   | Patched command | Type | Importance                      | Module            | ref. page |
|---|-----------------|------|---------------------------------|-------------------|-----------|
| ❸ | \@input         |      | important<br>but no side effect | interface-scrfile | 72        |

The \input primitive (saved by L^AT_EX in control sequence \@@input) is surrounded by \scr@load@hook{before}... \scr@load@hook{after}:

Original:

```
\def \@input#1{\IfFileExists{#1}
  { \@@input\@filef@und
    {\typeout{No file #1.}}%
  }%
```

Modified:

```
\renewcommand*\@input[1]{\IfFileExists{#1}
  { \scr@load@hook{before}{#1}%
    \@@input\@filef@und
    \scr@load@hook{after}{#1}%
    {\typeout{No file #1.}}%
  }%
```

### 20.2 † bookmark \\bookmark

|   | Patched command | Type  | Importance | Module             | ref. page |
|---|-----------------|-------|------------|--------------------|-----------|
| ❹ | \bookmark       | modif | important  | interface-bookmark | 41        |

The * keys for \bookmarksetup (italic*, color*, bold*, and disable*) are local to the group, but govern only the very next bookmark.

After the bookmark has been written, previous values must be restored at the end of each group until the same group level as the one in which the * option has been set is reached (if this group is the main group, level 0, then a global assignment is made).

Well... this is done at \write: \immediate\write{the bookmark entry} is replaced by \itf@BKM@write{the bookmark entry} to check first:

- 1) is the bookmark disable ?
- 2) is it necessary to restore previous values after write (*i.e.* \aftergroup) ?

## 20.3 † `colortbl`

|   | Patched command                                  | Type         | Importance                      | Module                          | ref. page |
|---|--------------------------------------------------|--------------|---------------------------------|---------------------------------|-----------|
| 5 | <code>\CT@cellc</code><br><code>\endarray</code> | <i>corr.</i> | important<br>but no side effect | <code>interface-makecell</code> | 66        |

`colortbl` modifies the preamble of tabulars: it put each cell in a box, sets the color and then displays the box. The example shown page 13 does not work without a minor modification: the color specification must be expanded short before their usage.

Original:

```
\def\CT@cellc#1[#2]#3{%
  \expandafter\gdef\expandafter\CT@cell@color\expandafter{%
    \expandafter\CT@color#1{[#2]}{#3}%
    \global\let\CT@cell@color\relax
  }{}}
```

Modified:

```
\def\CT@cellc#1[#2]#3{%
  #1 is either \@firstofone or \@gobble, [#2]#3 = color spec
  \xdef\CT@cell@color{\noexpand\CT@color#1{[#2]}{#3}%
  \global\let\noexpand\CT@cell@color\relax}{}}
```

`\endarray`: missed `\@arrayleft` in the redefinition of `colortbl`, then compatibility with `delarray` is broken. Bug fixed by `interfaces-base`.

Same problem for `arydshln` who redefines `\endarray`, taking care of `colortbl` but not of `delarray`. Bug fixed by `interfaces-base`.

Remember `arydshln` must be loaded after `colortbl` (this is not due to `interfaces`, but to the fact that both redefine `\@classz` from the `array` package).

## 20.4 † `hypbmsec`

|   | Patched command                                   | Type           | Importance                      | Module                                                               | ref. page |
|---|---------------------------------------------------|----------------|---------------------------------|----------------------------------------------------------------------|-----------|
| 6 | <code>\hbs@do</code><br><code>\hbs@process</code> | <i>prepend</i> | important<br>but no side effect | <code>interfaces-hypbmsec</code><br><code>interfaces-titlesec</code> | 44        |

In order to get `\lastsectiontitle`, `\lastsectionbookmark` and `\lastsectiontoc` updated at the right place (*i.e. after* `\sectionbreak` and *before* `\sectionmark`), some `interfaces`-macros have to be executed:

- at the beginning of `\hbs@do` → to get the word `section` or `subsection` etc..
- at the beginning of `\hbs@process` → to store the next section title, bookmark and toc

## 20.5 † `enumitem`

|   | Patched command       | Type         | Importance                          | Module                           | ref. page |
|---|-----------------------|--------------|-------------------------------------|----------------------------------|-----------|
| 7 | <code>\setlist</code> | <i>redef</i> | minor<br>optimisation or correction | <code>interfaces-enumitem</code> | 66        |

`\setlist` is redefined by `interfaces-enumitem` in order to extend the syntax.

## 20.6 † `pgfkeys.code.tex`

|   | Patched command                                                  | Type         | Importance | Module                          | ref. page |
|---|------------------------------------------------------------------|--------------|------------|---------------------------------|-----------|
| 8 | <code>\pgfkeys@unpack</code><br><code>\pgfkeys@case@three</code> | <i>redef</i> | important  | <code>interfaces-pgfkeys</code> | 74        |

`\pgfkeys@unpack` is modified to handle the case of the .blank handler: this is here that the value is conditionnally compared against .default, .blank or .value required.

`\pgfkeys@case@three` is modified to allow `handlers` to have .default, .blank or .value required values.

`\pgfkeys@exp@call`: missed a pair of braces around the value in `\pgfkeysalso`. Thus the code could break if the value contains an equal sign or a comma. Fixed.

# 21 Implementation

## 21.1 Interfaces

The package namespace is `itf@`.

```

1< *package>
2 \NeedsTeXFormat{LaTeX2e}[2005/12/01]
3 \ProvidesPackage{interfaces}
4   [2010/11/16 v2.2 - pgfkeys (pgf 2.0 et 2.1 macros for interfaces (FC)]

```

### 21.1.1 Requirements

```
5 \RequirePackage{interfaces-base}
```

### 21.1.2 The package options / list of interfaces

```

6 \DeclareOption{TeX}{\AtEndOfPackage{\LoadInterface{TeX}}}
7 \DeclareOption{marks}{\AtEndOfPackage{\LoadInterface{marks}}}
8 \def\itf@tp#1{%
9   \DeclareOption{#1}{\AtEndOfPackage{\LoadInterface{#1}}}%}
10 }% \itf@tp
11 \itf@tp{titlesec}
12 \itf@tp{hyperref}
13 \itf@tp{bookmark}
14 \itf@tp{fancyhdr}
15 \itf@tp{tocloft}
16 \itf@tp{makecell}
17 \itf@tp{enumitem}
18 \itf@tp{hypbmsec}
19 \itf@tp{environ}
20 \itf@tp{truncate}
21 \itf@tp{appendix}
22 \itf@tp{embedfile}
23 \itf@tp{fancyvrb}
24 \if\relax\@optionlist{interfaces.sty}\relax % no option given
25   \ExecuteOptions\@declaredoptions % all interfaces are active
26 \fi
27 \ProcessOptions
28</package>

```

## 21.2 Interfaces-base

```

29< *base>
30 \ProvidesPackage{interfaces-base}
31   [2010/11/16 v2.2 Basic macros for interfaces package (FC)]
32 \RequirePackage{etex,ltxcmds,etoolbox,pgfkeys,scrlfile}
33 \AtEndOfPackage{%
34   \RequirePackage{interfaces-etoolbox}%
35   \RequirePackage{interfaces-scrlfile}%
36   \RequirePackage{interfaces-pgfkeys}%
37   \RequirePackage{interfaces-tikz}%
38   \RequirePackage{interfaces-umrand}%
39 }% AtEndOfPackage

```

\LoadInterface Mechanic of `interfaces`: each module is loaded \AfterPackage.

```

40 \newrobustcmd*\LoadInterface[1]{%
41   \@for\itf@tp:=#1\do{%
42     \expandafter\ifx\csname\ifcsname\itf@tp\endcsname\else\fi\itf@tp\else\relax\fi\endcsname\marks
43     \itf@tp\else\relax\fi\endcsname\marks
44     \RequirePackage{interfaces-marks}%
45     \else\expandafter\ifx\csname\ifcsname\itf@tp\endcsname\else\fi\itf@tp\else\relax\fi\endcsname\LaTeX
46     \itf@tp\else\relax\fi\endcsname\LaTeX

```

```

47      \RequirePackage{interfaces-LaTeX}%
48      \else
49          \expandafter\itf@LoadInterface\expandafter{\itf@tp}%
50          \fi\fi}%
51 }% \LoadInterface
52 \@onlypreamble\LoadInterface
53 \newrobustcmd*\itf@LoadInterface[1]{\ltx@ifpackageloaded{interfaces-#1}%
54  {}%
55  {\AfterPackage*{#1}{\RequirePackage{interfaces-#1}}}%
56 }% \itf@LoadInterface

57 \newrobustcmd*\itf@Check@Package[1]{\ltx@ifpackageloaded{#1}%
58  {}%
59  {\PackageError{interfaces-#1}%
60      {The package ‘#1’ must be loaded before ‘interfaces-#1’}%
61      \MessageBreak Please refer to ‘interfaces.pdf’ document%
62      \MessageBreak for correct usage of the ‘interfaces’ package}\@ehd
63  }%
64 }% \itf@Check@Package

```

\itf@AfterPackage This is \AfterPackage* but it works inside document. `\ltxcmds` provides `\ltx@ifpackageloaded` to check if a package is loaded and – unlike L^AT_EX’s `\@ifpackageloaded` – this macro is not restricted to the document preamble.

```

65 \newrobustcmd*\itf@AfterPackage[1]{%
66   \csname\ltx@ifpackageloaded{#1}{\ltx@secondoftwo}{\@AfterPackage}\endcsname{#1}%
67 }% \itf@AfterPackage

```

### 21.2.1 Bug Fixes

David Carlisle’s `colortbl` is not compatible with David Carlisle’s `delarray`!!

Bug Fixed by `interfaces-base`. At the same time, the bug exist as well in `arydshln` and we fix it.

```

\endarray

68 \AfterPackage*{colortbl}{%
69  \renewcommand*\endarray{%
70    \crcr \egroup \egroup \arrayright \gdef\@preamble{}\CT@end}%
71 }% AfterPackage colortbl
72 \AfterPackage*{arydshln}{%
73  \ltx@ifpackageloaded{colortbl}%
74    \% colortbl
75    \def\endarray{%
76      \adl@endarray \egroup \adl@arrayrestore \CT@end \egroup \%<original>
77      \arrayright \% <FC>
78      \gdef\@preamble{}\% <FC>
79    }%
80  }%
81  \% no colortbl
82  \def\endarray{%
83    \adl@endarray \egroup \adl@arrayrestore \egroup \%<original>
84    \arrayright \% <FC>
85    \gdef\@preamble{}\% <FC>
86    }%
87  }%
88 }% AfterPackage arydshln

```

## 21.2.2 Interfaces patches

First a warning for loading `arydshln` after `colortbl`. Those packages does not interact properly.

```
89 \AfterPackage{colortbl}{%
90   {\ltx@ifpackageloaded{arydshln}%
91     {\PackageWarning{interfaces-base}%
92      {The package arydshln should be loaded after colortbl}%
93      \MessageBreak because both redefine \string@classz}}%
94   {}}%
95 }% AfterPackage colortbl
```

`\itf@preapptocmd` The need for this macro has rised to make `\itf@patch` simpler.

This is also more secure to do `\pretocmd` and `\apptocmd` in the same time, for the result (success or failure) is always the same for both.

```
96 \newrobustcmd*\itf@preapptocmd[5]{%
97   \pretocmd#1{#2}%
98   {\apptocmd#1{#3}%
99    {#4}%
100   {#5}}%
101   {#5}}%
102 }% \itf@preapptocmd
```

`\itf@patch` `\itf@patch` is used as a prefix to `\pretocmd`, `\apptocmd` and `\patchcmd`.

It's main purpose is to keep the information (`\PackageInfo`) in the case `\usepackage{interfaces}` has been put in a `\dump` format.

`\PackageInfo` is then deferred `\AtBeginDocument` time...

```
103 \def\itf@patch{@\makeother\#\futurelet\itf@tp\itf@patch@}
104 \def\itf@patch@{\csname itf@patch@\%
105   \ifx\itf@tp\patchcmd inside\else
106   \ifx\itf@tp\itf@preapptocmd inside\else
107   boundary\fi\fi
108   \endcsname
109 }% \itf@patch@
110 \def\itf@patch@boundary#1#2#3#4#5{%
111   \catcode`\#=6%
112   \let\itf@tp#2%
113   #1\itf@tp{#3}%
114   {\let#2\itf@tp
115     \AtBeginDocument{#4}% in order to keep this info even after \dump
116   }
117   {#5}}%
118 }% \itf@patch@boundary (for \pretocmd and \apptocmd)
119 \def\itf@patch@inside#1#2#3#4#5#6{%
120   \catcode`\#=6%
121   \let\itf@tp#2%
122   #1\itf@tp{#3}{#4}%
123   {\let#2\itf@tp
124     \AtBeginDocument{#5}%
125   }
126   {#6}}%
127 }% \itf@patch@inside (for \patchcmd)
```

### 21.2.3 Helper macros

```

128 \newskip\itf@sk
129 \newmuskip\itf@musk
130 \newtoks\itf@tk
131 \newcount\itf@cnt

\itf@strip@at \itf@strip@at removes all until @ while \itf@strip@dot removes the fractional part.
\itf@strip@dot

132 %% H E L P E R M A C R O S
133 \begingroup\makeatother
134   \csgdef{\itf@strip@at}{#1@{}}
135 \endgroup
136 \def\itf@strip@dot{#1.#2@nil{#1}}

```

\itf@expandonceifcs This macro is used to expand or not a comma separated list.

USAGE: \itf@expandonceifcs\macrotest

\macrotest is expanded once and its first token  $\tau$  is placed into a temporary macro \itf@tp.

If \macrotest equals \itf@tp: \macrotest is composed of that only token  $\tau$ , otherwise we leave and do nothing.

Then we test \noexpand $\tau$  against \relax with \ifcat. If false  $\tau$  cannot be a macro then we leave.

Then we test if  $\tau$  is expandable: \ifx $\tau$ =\expandonce $\tau$ ,  $\tau$  is not expandable and we leave. (may be \def $\tau$ { $\tau$ } but it's not *expandable*!).

Otherwise,  $\tau$  is expandable and \macrotest contains a single expandable token. Thus \macrotest shall be expanded once: \let\macrotest= $\tau$ .

```

137 \def\itf@expandonceifcs{\expandafter\itf@exp@ndonceifcs\expandafter#1#1Q@\nnil}
138 \def\itf@exp@ndonceifcs#1#2#3@\nnil{%
139   \def\itf@tp{#2}%
140   \ifx\itf@tp#1%
141     \ifcat\relax\noexpand#2%
142       \expandafter\expandafter\expandafter\remove@to@\nnil
143         \expandafter\ifx\expandafter#2#2@\nnil\else@\nnil\let#1#2\fi
144     \fi\fi
145 }% \itf@exp@ndonceifcs

```

### 21.2.4 Numbers

\itf@number \itf@number tries to get a number from its argument, and stores the result into the TeX count register given as first argument. Same syntax as L^AT_EX\@defaultunits.

A number is an integer that can be assigned to a count register, may be followed by one or more \relax.

```

146 %% G E T A N U M B E R (if possible)
147 %% USAGE: \itf@number\count<text>\relax@\nnil (like \@defaultunits)
148 \def\itf@number#1#2\relax@\nnil{%
149   \relax is to mimick the syntax of \@defaultunits
150   \afterassignment\itf@number#1\#2\relax\itf@number@%
151 }%
152 \def\itf@number@#1\relax#2\itf@number@{%
153   \ifblank{#1}{%
154     \csname ltx@\ifx\let@token\itf@number@@%
155       first\else second\fi oftwo\endcsname}%
156     \ltx@secondoftwo
157 }% \itf@number@
158 \def\itf@number@@#1{\futurelet\let@token\itf@number@@@}
159 \def\itf@number@@@{%
160   \csname \ifx\let@token\relax\itf@number@@%
161     remove@to@\nnil%
162   \fi\endcsname
163 }% \itf@number@@

```

## 21.2.5 Units and math units

`\itf@strip@mu` This is the equivalent of L^AT_EX's `\strip@pt` but for math units:

```
164 %% U N I T S F O R D I M E N S
165 \def\itf@strip@mu{\expandafter\itf@rem@mu\the}
166 {%
167 \@makeother\M\@makeother\U
168 \lowercase{%
169 \gdef\itf@rem@mu#1.#2MU{\#1\ifnum#2>\z@\#2\fi}%
170 }}%
```

`\itf@units@pttomu`  $1mu$  is exactly  $1em/18$  where the  $em$  is taken from `\textfont2`.

`\itf@units@muopt` Thus the scaling: `\fontdimen6\textfont2 * 100/1800`.

```
171 \% \skip = \itf@units@muopt \muskip
172 \def\itf@units@pttomu#1{\gluetomu\glueexpr
173   \glueexpr#1\relax*1800%
174     \expandafter\itf@strip@dot\the\dimexpr
175       \fontdimen6\textfont2*100\relax.\@nil
176     \relax
177 }% \itf@units@pttomu
178 \def\itf@units@muopt#1{\glueexpr
179   \mutoglue\muexpr#1\relax*
180     \expandafter\itf@strip@dot\the\dimexpr
181       \fontdimen6\textfont2*100\relax.\@nil/1800
182     \relax
183 }% \itf@units@muopt
```

`\itf@units@mu` Forces a dimension to be expressed in terms of math units ( $mu$ ), convert if needed. `\cftdotsep` in `tocloft` is expressed in math units.

```
184 \% \muskip = \itf@units@mu{ \skip or \muskip or number}
185 {%
186 \@makeother\P\@makeother\T\@makeother\S\@makeother\B
187 \@makeother\C\@makeother\M\@makeother\D\@makeother\I\@makeother\N
188 \lowercase{%
189 \gdef\itf@units@mu#1#2{\begingroup
190   \edef\itf@tp{#2}%
191   \atfor\x:={PT}{pt}{PC}{pc}{IN}{in}{BP}{bp}%
192     {CM}{cm}{MM}{mm}{DD}{dd}{CC}{cc}{SP}{sp}\do{%
193     \expandtwoargs\in@\x\itf@tp\ifin@\breakatfor\fi
194   }%
195   \expandafter\endgroup
196   \ifin@
197     #1=\itf@units@pttomu{#2}%
198   \else
199     \defaultunits#1 #2mu\relax\@nnil
200   \fi
201 }% \itf@units@mu
202 }}
```

`\itf@units@pt` Conversely, forces a unit to be expressed in terms of points. Convert (from  $mu$ ) if needed.

```
203 \% \skip = \itf@units@pt( \muskip or \skip or number )
204 {%
205 \@makeother\M\@makeother\U
206 \lowercase{%
207 \gdef\itf@units@pt#1#2{\begingroup
208   \expandtwoargs\in@{MU}{#2}%
209   \ifin@\else\expandtwoargs\in@{mu}{#2}\fi
210   \expandafter\endgroup
```

```

211   \ifin@
212     #1=\itf@units@mutopt{#2}%
213   \else
214     \@defaultunits#1 #2pt\relax\@nnil
215   \fi
216 }}}} \itf@units@pt

```

\itf@units@scale For multiplication of units: `\glueexpr...\relax*\itf@units@scale<scaling factor>`!

The scaling factor can be a fraction or a real number. Real numbers are approximated by a fraction of 65536.

```

217%% glue * 1.5 = glue * 98302 / 65536
218%% glue * 3/2 = glue * 3/2
219\def\itf@units@scale#1{\itf@units@scale@normalize#1/\@nnil/\@nil}
220\def\itf@units@scale@normalize#1/#2/#3@nil{%
221  \ifx#2\@nnil
222    \number\dimexpr#1pt\relax/65536
223  \else#1/#2
224  \fi
225 }% \itf@units@normalize

```

### 21.2.6 Default color

A trick about `\normalcolor`, which refers to `\default@color`:

`\default@color` is not expressed in terms of a color model. Therefore, it is not possible to convert it into a universally correct parameter for the `\color` command. That's why, for example, `\bookmark` ignore blank parameter for `color`: because it can do nothing with `\normalcolor` !

Solution: if a color named “`default`” has been defined in the preamble, then it is used as the default color. Otherwise, we define, `\AtBeginDocument` a color named “`default`”

If `xcolor` is loaded, an assignment: `\colorlet{default}{.}` is made `\AtBeginDocument`. If only `color` is loaded, then `\color{default}` means `\color{\default@color}` (this syntax is not admissible with `xcolor`, but it works with `color`).

```

226\AtBeginDocument{%
227  \ltx@ifpackageloaded{xcolor}%
228    {\@ifundefinedcolor{default}%
229      {\colorlet{default}{.}}%
230    {}%}
231    {\ltx@ifpackageloaded{color}%
232      {\@ifundefined{\string\color @default}%
233        {\expandafter\def\csname\string\color @default\endcsname{\default@color}}%
234      {}%}
235    {}%}
236 }
237</base>

```

## 21.3 Interface L^AT_EX

```

238<*LaTeX>
239\ProvidesPackage{interfaces-LaTeX}
240  [2010/10/31 v2.0 pgfkeys macros for LaTeX (FC)]
241\RequirePackage{interfaces-base}
242\itf@pgfkeys

```

### 21.3.1 \changefont

\changefont First use of `\pgfkeys` handlers `./store as glue in cs`, `./estore as dimen in cs`... The handlers create the `+` – and `*` keys automatically. As far as the `+` – and `*` keys invoke the main key, it is possible to `./append code` to the main key, without taking care of the `+` – and `*` keys.

The `./alias*` handler creates the aliases for the main key and the `+` `-` and `*` keys in one shot.

```

243 \newrobustcmd*\changefont[1]{\itf@pgfkeys
244   \pgfqkeys{/itf/TeX/changefont}{#1}\selectfont
245 }% \changefont
246 \AfterPackage*{hyperref}
247   {\pdfstringdefDisableCommands{\let\changefont\gobble}}


248 \pgfqkeys{/itf/TeX/changefont}{%
249   encoding/.code=\fontencoding{#1},encoding/.value required,%
250     enc/.alias=encoding,%
251   family/.code=\fontfamily{#1},family/.value required,%
252     fam/.alias=family,%
253   series/.code=\fontseries{#1},series/.value required,%
254     ser/.alias=series,%
255   shape/.code=\fontshape{#1},shape/.value required,%
256     sha/.alias=shape,%
257   size/.estore as dimen in cs=f@size,%
258   size/.append code=\set@fontsize\f@linespread\f@size\f@baselineskip,%
259     siz/.alias*=size,%
260   baselineskip/.store as glue in cs=f@baselineskip,%
261   baselineskip/.append code=\set@fontsize\f@linespread\f@size\f@baselineskip,%
262     skip/.alias*=baselineskip,%
263   linespread/.estore as dimen in cs=f@linespread,%
264   linespread/.add code={\ifx\f@linespread\empty\let\f@linespread\ne\fi}%
265     {\edef\f@linespread{\expandafter\rem@pt\f@linespread}%
266      \set@fontsize\f@linespread\f@size\f@baselineskip},%
267   linespread+/.prefix code=\ifx\f@linespread\empty\let\f@linespread\ne\fi,%
268   linespread-/.prefix code=\ifx\f@linespread\empty\let\f@linespread\ne\fi,%
269   linespread*/.prefix code=\ifx\f@linespread\empty\let\f@linespread\ne\fi,%
270     spread/.alias*=linespread,%
271 }% \pgfkeys definitions for \changefont
272 \itf@AfterPackage{color}{%
273   \pgfkeysifdefined{/itf/TeX/changefont/color/.@cmd}{}%
274   {\pgfkeys{/itf/TeX/changefont/color/.code=\color{#1}}}%
275 }% AfterPackage color
276 \itf@AfterPackage{xcolor}{%
277   \pgfkeysifdefined{/itf/TeX/changefont/color/.@cmd}{}%
278   {\pgfkeys{/itf/TeX/changefont/color/.code=\color{#1}}}%
279 }% AfterPackage xcolor

280 
```

`280`

## 21.4 Interface TikZ ( pgfmanual )

```

281 {*tikz}
282 \ProvidesPackage{interfaces-tikz}
283   [2010/10/31 v2.0 load tikz libraries easily - pgf 2.0 & 2.1 (FC)]
284 \RequirePackage{interfaces-base}
285 \itf@pgfkeys

```

### 21.4.1 Catcode verifications

Some languages (french) changes the catcode of punctuations symbols: `:` and `!` are active, and this is bad for tikz calculations.

`\tikzset{every picture/.style=@makeother...}` solves the problem.

```

286 %% Some catcode checking
287 \AfterPackage*{tikz}{%
288   \AtEndPreamble{%
289     \tikzset{every picture/.add style=%
290       /utils/exec={@\makeother\!@\makeother\!:}}{}%
291   }%
292 }%

```

## 21.4.2 List of pgf/TikZ packages and libraries

`\alltikzpackages` Record “modules” that must be loaded with `\usepackage`:

```
293 \newcommand*\alltikzpackages{%
294   tikz,pgf,pgfcore,pgfcalendar,pgffor,pgfkeys,pgfpages,pgfrcs,tikzexternal,xxcolor,%
295   pgfplots,pgfsys,pgfmath,%
296   pgfpic2e,fp,%           + fp required for fixedpointarithmetic
297 }% \alltikzpackages
```

`\allpgfmodules` Record modules that must be loaded with `\usepgfmodule`:

```
298 \newcommand*\allpgfmodules{datavisualization,%
299 decorations,%
300 matrix,%
301 oo,%
302 parser,%
303 plot,%
304 shapes,%
305 snakes,%
306 sorting%
307 }% \allpgfmodules
```

`\alltikzlibraries` Record modules that must be loaded with `\usetikzlibrary`.

If a module is not found in any of the three lists `\alltikzpackage`, `\allpgfmodules`, and `\alltikzlibraries` then `\usetikzlibrary` is used by default.

```
308 \newcommand*\alltikzlibraries{3d,%
309 arrows,%
310 automata,%
311 backgrounds,%
312 calc,%
313 calendar,%
314 chains,%
315 circuits,%
316 circuits.ee,%
317 circuits.ee.IEC,%
318 circuits.logic,%
319 circuits.logic.CDH,%
320 circuits.logic.IEC,%
321 circuits.logic.US,%
322 datavisualization,%
323 datavisualization.3d,%
324 datavisualization.barcharts,%
325 datavisualization.formats.functions,%
326 datavisualization.polar,%
327 datavisualization.sparklines,%
328 decorations,%
329 decorations.footprints,%
330 decorations.fractals,%
331 decorations.markings,%
332 decorations.pathmorphing,%
333 decorations.pathreplacing,%
334 decorations.shapes,%
335 decorations.text,%
336 er,%
337 external,%
338 fadings,%
339 fit,%
340 fixedpointarithmetic,%
341 folding,%
342 fpu,%
343 graphs,%
```

```

344 graphs.basic,%
345 intersections,%
346 lindenmayersystems,%
347 matrix,%
348 mindmap,%
349 patterns,%
350 petri,%
351 plothandlers,%
352 plotmarks,%
353 positioning,%
354 scopes,%
355 shadings,%
356 shadows,%
357 shapes,%
358 shapes.arrows,%
359 shapes.callouts,%
360 shapes.gates.logic.IEC,%
361 shapes.gates.logic.US,%
362 shapes.geometric,%
363 shapes.misc,%
364 shapes.multipart,%
365 shapes.symbols,%
366 snakes,%
367 spy,%
368 svg.path,%
369 through,%
370 topaths,%
371 trees,%
372 turtle%
373 }% \alltikzlibraries

```

### 21.4.3 pgfkeys definitions for \usetikz

```

374 \pgfqkeys{/itf/tikz}{%
375   basic/.code={%
376     \usetikzlibrary{calc,%
377                   arrows,%
378                   backgrounds,%
379                   scopes}%
380   }%
381 },%
382 basic/.value forbidden,% \alltikzpackages
383 base/.alias=basic,%
384 all/.code={%
385   \expandafter\RequirePackage\expandafter{\alltikzpackages}%
386   \expandafter\usepgfmodule\expandafter{\allpgfmodules}% see pgfcorescopes.code.tex
387   \expandafter\usetikzlibrary\expandafter{\alltikzlibraries}% from: tikz.code.tex
388 },%
389 .unknown/.code=\itf@usetikzcode\pgfkeyscurrentname
390 }% \pgfkeys for \usetikz

```

\usetikz

```

391 \newrobustcmd*\usetikz[1]{%
392   \RequirePackage{tikz}%
393   \ifblank{#1}%
394     {\pgfqkeys{/itf/tikz}{all}}%
395     {\pgfqkeys{/itf/tikz}{#1}}%
396   \itf@pgfkeys% overwritten by \input primitive otherwise
397 }% \usetikz

```

\itf@usetikzcode For unknown keys: when a library is referenced.

```

398 \newrobustcmd*\itf@usetikzcode[1]{\begingroup
399   \edef\x{%
400     \noexpand\in@{,#1}{,\alltikzpackages,}%
401     \noexpand\ifin@ \noexpand\usepackage{#1}%
402     \noexpand\else
403       \noexpand\in@{,#1}{,\allpgfmodules,}%
404       \noexpand\ifin@ \noexpand\usepgfmodule{#1}%
405       \noexpand\else \noexpand\usetikzlibrary{#1}\usetikzlibrary by default
406       \noexpand\fi
407     \noexpand\fi
408   }\expandafter\endgroup\x
409 }% \itf@usetikzcode

410 
```

## 21.5 Interface titlesec : \sectionformat

```

411 {*titlesec}
412 \ProvidesPackage{interfaces-titlesec}
413 [2010/11/15 v2.1 pgfkeys macros for titlesec (FC)]
414 \RequirePackage{interfaces-base}
415 \RequirePackage{auxhook}% for fix for \ttlp@append (odd/even pages, not implemented in interf
416 \itf@Check@Package{titlesec}
417 \itf@pgfkeys

```

### 21.5.1 titlesec modifications

```

418%% t i t l e s e c "p a t c h e s"
419 \robustify\titleclass \robustify\titleformat \robustify\titlespacing
420 \AddLineBeginMainAux{%
421   \string\providetoggle\string\ttlp@append[2]{}}%
422 }
423 \BeforeFile*{ttlkeys.def}{%
424   \catcode32 9\relax % ignore spaces
425   \count@\endlinechar\endlinechar\m@ne % and line feeds
426 }
427 \AfterFile*{ttlkeys.def}{% restore spaces and line feeds
428   \catcode32 10\relax
429   \endlinechar\count@
430 }

```

\ttlh@runinifempty A “new” shape which behaves like `hang` but no paragraph break is inserted if the title is empty. Hence the name: **runinifempty**.

```

431 %-----%
432 %% runinifempty => like hang but no paragraph break (like runin) if title is empty
433 \def\ttlh@runinifempty{%
434   \csname ttlh@\ifx\ttl@savetitle\@empty runin\else hang\fi\endcsname
435 }% \ttlh@runinifempty

```

### 21.5.2 \sectionformat

\sectionformat \sectionformat\subsection* [<shape>]{<key=value>}

```

436 \newrobustcmd*\sectionformat[1]{\ifstar
437   {\itf@sectionformat{#1}{/*}}
438   {\itf@sectionformat{#1}{}}%
439 }% \sectionformat
440 \def\itf@sectionformat#1#2{@testopt
441   {\expandafter\expandafter\expandafter%
442    \itf@sectionf@rmat\expandafter\expandafter\gobble\string#1\@nil{#2}}{}%
443 }% \sectionformat
444 \def\itf@sectionf@rmat#1\@nil#2[#3]#4{\itf@pgfkeys

```

```

445   \ifblank{#3}
446     {\pgfqkeys{/itf/titlesec/#1#2}
447      {/itf/titlesec/@setup={#1}{#2},#4,/itf/titlesec/@update={#1}{#2}}}
448     {\pgfqkeys{/itf/titlesec/#1#2}
449      {/itf/titlesec/@setup={#1}{#2},#4,shape={#3},/itf/titlesec/@update={#1}{#2}}}}
450 }% \itf@ttl@sectionformat

```

`\itf@ttl@setup` Macros used to retrieve the values set by `\titleformat` and/or `\titlespacing`. This allows to use `\itf@ttl@getformat` `\sectionformat` concurrently with `\titleformat` and `\titlespacing` provided by `titlesec`.

```

\itf@ttl@getspacing 451% private keys @setup and @update
452 \pgfqkeys{/itf/titlesec}{%
453   @setup/.code 2 args=\itf@ttl@setup{#1}{#2},%
454   @update/.code 2 args=\expandafter\itf@ttl@update
455   \csname#1\endcsname{/itf/titlesec/#1}{#2}%
456 }
457 \newrobustcmd*\itf@ttl@setup[2]{%
458   \pgfkeysifdefined{/itf/titlesec/#1#2/shape}{}%
459   {\itf@ttl@newsection{#1}{#2}}%
460   \ifcsundef{ttls@#1#2}{}% no previous spacing settings
461   {\expandafter\expandafter\expandafter%
462    \itf@ttl@getspacing\csname ttls@#1#2\endcsname{/itf/titlesec/#1#2}}%
463   \ifcsundef{ttlf@#1#2}{}% no previous format settings
464   {\expandafter\expandafter\expandafter%
465    \itf@ttl@getformat\csname ttlf@#1#2\endcsname{/itf/titlesec/#1#2}}%
466 }% \itf@ttl@setup
467 \def\itf@ttl@setkey#1#2{\begingroup
468   \def\x{\pgfkeysvalueof{#1}}\def\y{#2}%
469   \expandafter\endgroup\ifx\x\y\else
470   \pgfkeyssetvalue{#1}{#2}\fi
471 }% \itf@ttl@setkey
472 \def\itf@ttl@getspacing#1#2#3#4#5#6{%
473   \itf@ttl@setkey{#6/left}{#1}%
474   \itf@ttl@setkey{#6/right}{#2}%
475   \itf@ttl@setkey{#6/top}{#3}%
476   \itf@ttl@setkey{#6/bottom}{#4}%
477   \itf@ttl@setkey{#6/indent first}{#5}%
478 }% \itf@ttl@getspacing
479 \def\itf@ttl@getformat#1#2{%
480   \csname ltx@\ifx\pgfkeysvalueof#1first\else second\fi oftwo\endcsname
481   {\itf@ttl@get@format{}}
482   {\itf@ttl@get@format{#1}{#2}}%
483 }% \itf@ttl@getformat
484 \def\itf@ttl@get@format#1#2#3#4#5#6#7{%
485   \ifrelax\detokenize{#1}\relax\else
486   \pgfkeysEsetvalue{#7/shape}{\expandafter\itf@strip@at\string#1}%
487   \fi
488   \itf@ttl@setkey{#7/font}{#2}%
489   \itf@ttl@setkey{#7/label}{#3}%
490   \itf@ttl@setkey{#7/sep}{#4}%
491   \itf@ttl@setkey{#7/before}{#5}%
492   \itf@ttl@setkey{#7/after}{#6}%
493 }% \itf@ttl@get@format

```

`\itf@ttl@update` This macros plugs the values set before with `\pgfkeys` into the special macros `\ttlf@<section>` (for format) and `\ttls@<section>` (for spacing).

```

494 \newrobustcmd*\itf@ttl@update[3]{%
495   \ifblank{#3}
496     {\ttl@getkeys{#1}{titlesec}}
497     {\ttl@getkeys{name=#1,numberless}{titlesec}}%
498   \pgfkeysgetvalue{#2#3/shape}\itf@tp
499   \ifcsundef{ttlh@\itf@tp}

```

```

500      {\begingroup \makeatletter
501       \InputIfFileExists{\itf@tp.tss}{}{%
502         \PackageError{titlesec}{Unknown shape}{%
503           {Shapes are defined in files with extension tss\MessageBreak
504             Either you have misspelled the shape\MessageBreak
505             or there is no a \itf@tp.tss file}}%
506       \endgroup
507     }%
508   \ifttl@explicit@\temptokena{\csdef{ttl@}{\ttl@a}##1}%
509   \else@\temptokena{\csdef{ttl@}{\ttl@a}}%
510   \fi
511   \the@\temptokena{%
512     \pgfkeysvalueof{#2#3/shape@cs}%
513     \pgfkeysvalueof{#2#3/font}%
514     \pgfkeysvalueof{#2#3/label}%
515     \pgfkeysvalueof{#2#3/sep}%
516     \pgfkeysvalueof{#2#3/before}%
517     \pgfkeysvalueof{#2#3/after}%
518   \csname ttl@compat\ttl@a\endcsname
519   \ttl@spacing@ii
520     \pgfkeysvalueof{#2#3/indent first}%
521     \pgfkeysvalueof{#2#3/left}%
522     \pgfkeysvalueof{#2#3/top}%
523     \pgfkeysvalueof{#2#3/bottom}%
524     \pgfkeysvalueof{#2#3/right}]%
525 }% \itf@ttl@update

```

\itf@ttl@titleclass This macro calls \titleclass to either:

- change the class of an existing sectioning command;
- or:
- create a new sectioning command.

```

526 %%-----%
527 %%  t i t l e c l a s s
528 \newrobustcmd*\itf@ttl@titleclass{\ifstar
529   {\itf@ttl@titleclass@i*}
530   {\itf@ttl@titleclass@i{}}
531 }% \itf@ttl@titleclass
532 \def\itf@ttl@titleclass@i#1#2[#4]#5@nil{\ifblank{#4}
533   {%
534     \expandafter\titleclass\csname#2\endcsname{#3}%
535   }
536   {%
537     \expandafter\titleclass\csname#2\endcsname{#3}[{#4}]% #4 = sup-level
538     \if\relax#1\relax % not starred
539       \expandafter\expandafter\expandafter%
540         \itf@ttl@titleclass@ii\expandafter\gobble\string#4\relax{#2}% subsection\relax{%
541     \fi
542   }%
543   \expandafter\providecommand\csname l@#2\endcsname[2]{% no entry in toc
544 }% \itf@ttl@titleclass@i
545 \def\itf@ttl@titleclass@ii#1\relax#2{%
546   \ifcsundef{c@#2}
547     {\newcounter{#2}[{#1}]}% creates counter for #1
548     {\@addtoreset{#2}{#1}}%
549   \csxdef{the#2}{%
550     \expandafter\noexpand\csname the#1\endcsname
551     .\noexpand\arabic\expandafter\noexpand\csname c@#2\endcsname
552   }% \the#2
553   \ifcsundef{l@#1}{}{%
554     \expandafter\providecommand\csname l@#2\endcsname[2]{%
555       \csname l@#1\endcsname{##1}{##2}%
556     }% \l@#2 = \l@newsection is (by default) the same as \l@sup-level

```

```
557    }%
558 }% \itf@ttl@titleclass@ii
```

\itf@ttl@createsection If `\sectionformat` is invoked with an unknown sectionning command (*i.e.* an undefined macro as first argument) then `\itf@ttl@createsection` is expanded in order to:

- 1) create the new sectionning command with `\titleclass`,
- 2) create the `pgfkeys` keys for this section,
- 3) do some additionnal setup if `interfaces-hypbmsec` and/or `interfaces-tocloft` are loaded.

```
559 %%-----%
560 %% Creating new sections
561 \def\itf@ttl@createsection#1{%
  sectionning command undefined: temporarily defined
  \expandafter\titleclass\csname#1\endcsname{straight}[*]%
  in straight class
562   \itf@AfterPackage{hypbmsec}{%
563     \AfterPreamble{%
564       \csletcs{hbs@#1}{#1}%
565       \csxdef{#1}{\noexpand\hbs@do\expandafter\noexpand\csname hbs@#1\endcsname}%
566     }%
567   }%
568 }%
569 \itf@AfterPackage{interfaces-tocloft}{%
570   \itf@cft@newsection{#1}{toc}%
571 }%
572 \itf@AfterPackage{interfaces-hypbmsec}{%
573   \expandafter\let\csname last#1title\endcsname\relax
574   \expandafter\let\csname last#1bookmark\endcsname\relax
575   \expandafter\let\csname last#1toc\endcsname\relax
576 }%
577 }% \itf@ttl@createsection
```

### 21.5.3 Keys for `\sectionformat`

\itf@ttl@newsection This macro creates the `pgfkeys` keys for each sectionning command.

The `pgfkeys` keys are not created at once, but only when `\sectionformat` is invoked first.

```
578 %%-----%
579 %% Defining keys for each section
580 \def\itf@ttl@newsection#1#2{%
  #1=section, #2=/* or empty
  \ifcscundef{#1}{%
    sectionning command undefined: temporarily defined in straight class
    \itf@ttl@createsection{#1}%
  }%
  \ifblank{#2}{%
    \pgfqkeys{/itf/titlesec/#1}{%
      these keys do not exist for starred/numberless sections
      class/.code=\itf@ttl@titleclass{#1}##1[]\@nil,% with \@addtoreset
      class/.value required,% if sup-level is given
      class*/.code=\itf@ttl@titleclass*{#1}##1[]\@nil,% without \@addtoreset
      class*/.value required,% even if sup-level is given
      shape/.initial=,% initial shapes set by titlesec.sty for not numbered sections.
      the#1/.store in cs=the#1,%
      the#1/.default=\pgfkeysvalueof{/itf/titlesec/#1/label},%
    }%
    % starred (numberless) unless titlesec, we do not put the default shape to hang,
    \pgfkeysifdefined{/itf/titlesec/#1/shape}{%
      but to the shape of the not starred
      \pgfkeysgetvalue{/itf/titlesec/#1/shape}\itf@tp% sectionning command
      \def\itf@tp{hang}}%
    \pgfkeyset{/itf/titlesec/#1#2/shape}\itf@tp%
  }%
  \pgfqkeys{/itf/titlesec/#1#2}{%
    shape/.value required,%
    shape@cs/.initial=
    \ifdefined\itf@last@from@next\itf@last@from@next{#1}\fi % interfaces-hypbmsec
  }%
}
```

```

604     \csname ttlh@\pgfkeysvalueof{/itf/titlesec/#1#2/shape}\endcsname,% constant
605     shape@cs/.is private=interface-titlesec,% constant
606     font/.store as code,%  

607     format/.alias*=font,%  

608     label/.store as code,%  

609     sep/.estore as glue,%  

610     labelsep/.alias*=sep,%  

611     before/.store as code,%  

612     after/.store as code,%  

613     indent first/.true=\pgfkeyssetvalue{/itf/titlesec/#1#2/indent first}{\@ne},%  

614     indent first/.false=\pgfkeyssetvalue{/itf/titlesec/#1#2/indent first}{\z@},%  

615     identfirst/.alias=indent first,%  

616     indent first=true,%  

617     left/.estore as glue,%  

618     top/.estore as glue,%  

619     bottom/.estore as glue,%  

620     right/.estore as glue,%  

621     break/.code=\ifblank{##1}  

622         {\cslet{\#1break}{\empty}}%  

623         {\protected\csdef{\#1break}{##1}},%  

624     break*/.code=\protected\apptocs{\#1break}{##1},%  

625     break+/.alias=break*,%  

626     mark/.code=\ifblank{##1}  

627         {\cslet{\#1mark}{\gobble}}%  

628         {\protected\long\csdef{\#1mark}{{\relax\iffalse\relax\#1\relax\#1}}},%  

629     mark/.default=%  

630 }% \pgfkeys  

631 \ifblank{#2}{% no default marks for starred (numberless) \section*  

632 \itf@AfterPackage{interfaces-marks}{% requires interfaces-marks  

633     \pgfkeys{/itf/titlesec/#1/mark/.default=\marksthe{#1}{##1}}%  

634 }% AfterPackage interfaces-mark  

635 \itf@AfterPackage{interfaces-bookmark}{% requires interfaces-bookmark  

636 \pgfqkeys{/itf/titlesec/#1}{% does not exist for starred sections  

637     bookmark/.is multiple choice,%  

638     bookmark/level/.estore in cs=toclevel@#1,% depth of the bookmark  

639     bookmark/level/.default=\ifcsvoid{ttll@#1}{99}{\csuse{ttll@#1}},%  

640     bookmark/level/.blank=\ifcsvoid{toclevel@#1}{\ifcsvoid{ttll@#1}{99}{\csuse{ttll@#1}}}{},%  

641     bookmark/toclevel/.alias=bookmark/level,%  

642     bookmark/bold/.code=\ifblank{##1}  

643         {\pgfkeyslet{/itf/titlesec/#1/bookmark/bold}{\empty}}%  

644         {\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/bold}{bold=##1}},%  

645     bookmark/bold/.default=true,%  

646     bookmark/bold*/.code=\ifblank{##1}  

647         {\pgfkeyslet{/itf/titlesec/#1/bookmark/bold*}{\empty}}%  

648         {\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/bold*}{bold*=##1}},%  

649     bookmark/bold*/.default=true,%  

650     bookmark/italic/.code=\ifblank{##1}  

651         {\pgfkeyslet{/itf/titlesec/#1/bookmark/italic}{\empty}}%  

652         {\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/italic}{italic=##1}},%  

653     bookmark/italic/.default=true,%  

654     bookmark/italic*/.code=\ifblank{##1}  

655         {\pgfkeyslet{/itf/titlesec/#1/bookmark/italic*}{\empty}}%  

656         {\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/italic*}{italic*=##1}},%  

657     bookmark/italic*/.default=true,%  

658     bookmark/color/.code=\ifblank{##1}  

659         {\pgfkeyslet{/itf/titlesec/#1/bookmark/color}{\empty}}%  

660         {\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/color}{color=##1}},%  

661     bookmark/color/.default=%  

662     bookmark/color*/.code=\ifblank{##1}  

663         {\pgfkeyslet{/itf/titlesec/#1/bookmark/color*}{\empty}}%  

664         {\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/color*}{color*=##1}},%  

665     bookmark/color*/.default=%  

666     bookmark/on/.code=\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/on}{disable=false},%  

667     bookmark/on/.value forbidden,%  

668     bookmark/off/.code=\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/on}{disable=true},%

```

```

669     bookmark/off/.value forbidden,%
670     bookmark/none/.alias=bookmark/off,%
671     bookmark/style/.code=\ifblank{##1}%
672         {\pgfkeyslet{/itf/titlesec/#1/bookmark/style}\@empty}
673         {\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/style}{style=##1},}%,%
674     bookmark/style/.default=,%
675     bookmark/.unknown/.code=\pgfkeysEsetvalue{/itf/titlesec/#1/bookmark/other}{\pgfkeys}
676     bookmark/bold/.initial=,%
677     bookmark/bold*/.initial=,%
678     bookmark/italic/.initial=,%
679     bookmark/italic*/.initial=,%
680     bookmark/color/.initial=,%
681     bookmark/color*/.initial=,%
682     bookmark/on/.initial=,%
683     bookmark/style/.initial=,%
684 }% \AfterPackage*{interfaces-bookmark}
685 }% not starred (numberless) \section
686 {}% starred/numberless \section*: do not create the keys
687 }% \itf@ttl@newsection

```

\addcontentsline (patch) This modification of \addcontentsline occurs only if both **interfaces-bookmark** and **interfaces-titlesec** are loaded. Implementation of the **bookmark** key and sub-keys of \sectionformat for each sectionning command.

```

688 %% bookmarks styles for sectionning commands:
689 \@makeother\#
690 \itf@AfterPackage{interfaces-bookmark}{%
691     \itf@patch\itf@preapptocmd\addcontentsline % #2=section name in
692     {\begingroup\itf@ttl@BKM{/itf/titlesec/#2/bookmark}}% \addcontentsline
693     {\endgroup}
694     {\PackageInfo{interfaces-titlesec}{%
695         {Command \string\addcontentsline\space successfully patched!}}
696     {\PackageWarning{interfaces-titlesec}{%
697         {Patch of \string\addcontentsline\space failed
698         \MessageBreak bookmark-style specs for sections
699         \MessageBreak will be unavailable}}}}
700 }% AfterPackage interfaces-bookmark
701 \catcode`\#=6
702 \def\itf@ttl@BKM#1{%
703     \pgfkeysifdefined{#1/.@cmd}{%
704         {%
705             \edef\itf@tp{\% do nothing if the keys are not set (shortcut)
706             \pgfkeysvalueof{#1:bold}%
707             \pgfkeysvalueof{#1:bold*}%
708             \pgfkeysvalueof{#1/italic}%
709             \pgfkeysvalueof{#1/italic*}%
710             \pgfkeysvalueof{#1/color}%
711             \pgfkeysvalueof{#1/color*}%
712             \pgfkeysvalueof{#1/on}%
713             \pgfkeysvalueof{#1/style}}%
714             \ifx\itf@tp\@empty
715             \else
716                 \global\pgfkeyslet{#1:bold*}\@empty
717                 \global\pgfkeyslet{#1/italic*}\@empty
718                 \global\pgfkeyslet{#1/color*}\@empty
719                 \expandafter\bookmarksetup\expandafter{\itf@tp}%
720                 \pgfkeysifdefined{/itf/bookmark/*}{%
721                     {%
722                         \bookmarksetup{color*=} overwrites \sectionformat...{bookmark/color*=}
723                         \pgfkeysgetvalue{/itf/bookmark/*}\itf@tp
724                         \expandafter\bookmarksetup\expandafter{\itf@tp}%
725                     }
726                 }%
727             \fi
728         }%
729     }%
730 }

```

```
728 }% \itf@ttl@BKM
```

#### 21.5.4 \bottomtitles

\bottomtitles This is a “skip command”: it can be assigned a number or a skip.

```
729 %-----  
730 %% b o t t o m t i t l e s  
731 \newrobustcmd*\bottomtitles{\ifnextchar\bgroup  
732   \itf@setbottomtitlespace  
733   {\afterassignment\itf@set@bottomtitle@space\itf@sk}%  
734 }% \bottomtitles  
735 \def\itf@setbottomtitlespace#1{  
736   @defaultunits\itf@sk#1\textheight\relax@nnil  
737   \itf@set@bottomtitle@space  
738 }% \itf@setbottomtitlespace  
739 \def\itf@set@bottomtitle@space{\edef\bottomtitlespace{\the\itf@sk}}  
  
740 </titlesec>
```

### 21.6 + Interface ↪ hyperref

```
741 {*hyperref}  
742 \ProvidesPackage{interfaces-hyperref}  
743   [2010/10/27 v1.2 provides raised hyperdef (FC)]  
744 \RequirePackage{interfaces-base}  
745 \itf@Check@Package{hyperref}
```

\raisedhyperdef The anchor is made with \Hy@raisedlink. The raised amount is used in \HyperRaiseLinkHook:

```
746 \newrobustcmd\raisedhyperdef[4][]{\begingroup  
747   \ifblank{#1}{  
748     {}  
749     {}%  
750     \expandafter\def\expandafter\HyperRaiseLinkHook\expandafter{  
751       \expandafter\def\expandafter\HyperRaiseLinkHook  
752         \expandafter{\HyperRaiseLinkHook}% restore current definition  
753       \HyperRaiseLinkHook  
754       \advance\noexpand\HyperRaiseLinkLength by\dimexpr#1\relax\relax}%  
755     }%  
756   \ifx \\#2\\def \Hy@AnchorName {#3}\else  
757     \def \Hy@AnchorName {#2.#3}%  
758   \fi  
759   \ltx@leavevemode  
760   \Hy@raisedlink{\hyper@anchorstart{\Hy@AnchorName}\hyper@anchorend}%  
761   \endgroup  
762   #4%  
763 }% \raisedhyperdef  
  
764 </hyperref>
```

## 21.7 + Interface ↗ bookmark

```

765 /*bookmark*/
766 \ProvidesPackage{interfaces-bookmark}
767   [2010/11/15 v2.1 more control on bookmarks (FC)]
768 \RequirePackage{interfaces-base}
769 \RequirePackage{interfaces-hyperref}
770 \RequirePackage{refcount}
771 \itf@Check@Package{bookmark}
772 \itf@pgfkeys

```

\bookmark (patch) Implementation of `disable`, and starred keys `*` (previous values must be restored after `\write`):

```

773 \expandafter\itf@patch\expandafter\patchcmd\csname string\bookmark\endcsname
774   {\immediate\write\mainaux}{\itf@BKM@write}
775   {\PackageInfo{interfaces-bookmark}
776     {Command \expandafter\string\csname string\bookmark\endcsname\space
777      successfully patched!}}
778   {\PackageWarning{interfaces-bookmark}
779     {Command \expandafter\string\csname string\bookmark\endcsname\space
780      could not be patched\MessageBreak
781      Loading of interfaces-bookmark is canceled}%
782   \endinput}

```

### 21.7.1 star keys* definitions

```

783 \global\let\itf@BKM@enable@ne 0 = disable*, 1=enable, @M = disable
784 \pgfqkeys{/itf/bookmark}{%
785   disable/.true=\global\let\itf@BKM@enable@M,%
786   disable/.false=\global\let\itf@BKM@enable@ne,%
787   disable*/.true=\global\let\itf@BKM@enable@z@,%
788   disable*/.false=\global\let\itf@BKM@enable@ne,%
789   atend/.true=\ifnum\itf@BKM@enable>0\edef\itf@BKM@enable{\number-\itf@BKM@enable}\fi,%
790   atend/.false=\ifnum\itf@BKM@enable<0\edef\itf@BKM@enable{\number-\itf@BKM@enable}\fi,%
791   bold*/.true=\pgfqkeys{/itf/bookmark}{@save={bold=true}},%
792   bold*/.false=\pgfqkeys{/itf/bookmark}{@save={bold=false}},%
793   bold*/@blank/.code=\pgfqkeys{/itf/bookmark/@restore}{bold},%
794   bold*/.blank=@blank,%
795   italic*/.true=\pgfqkeys{/itf/bookmark}{@save={italic=true}},%
796   italic*/.false=\pgfqkeys{/itf/bookmark}{@save={italic=false}},%
797   italic*/@blank/.code=\pgfqkeys{/itf/bookmark/@restore}{italic},%
798   italic*/.blank=@blank,%
799   color*/.code={\ifblank{#1}{%
800     {\pgfqkeys{/itf/bookmark/@restore}{color}}%
801     {\pgfqkeys{/itf/bookmark}{@save={color={#1}}}}%
802   },%
803   @save/.code={%
804     \pgfkeysifdefined{/itf/bookmark/@save/color}{}%
805       {\pgfkeyslet{/itf/bookmark/@save/color}\BKM@color}%
806     \pgfkeysifdefined{/itf/bookmark/@save/FontStyle}{}%
807       {\pgfkeyslet{/itf/bookmark/@save/FontStyle}\BS@BKM@FontStyle}%
808     \pgfkeysifdefined{/itf/bookmark/@save/grouplevel}{}%
809       {\pgfkeysEsetvalue{/itf/bookmark/@save/grouplevel}{\the\currentgrouplevel}}%
810     \pgfkeysaddvalue{/itf/bookmark/*}{}{, #1}%
811     \bookmarksetup{#1}%
812   },%
813   @restore/color/.code=%
814     \pgfkeysifdefined{/itf/bookmark/@save/color}%
815       {\pgfkeysgetvalue{/itf/bookmark/@save/color}\BKM@color}%
816     {},%
817   @restore/FontStyle/.code=%
818     \pgfkeysifdefined{/itf/bookmark/@save/FontStyle}%
819       {\pgfkeysgetvalue{/itf/bookmark/@save/FontStyle}\BS@BKM@FontStyle}%

```

```

820      {} ,%
821      @restore/.code={%
822 %%      \pgfqkeys{/itf/bookmark/@restore}{color,FontStyle}%
823      % repeating the following lines
824      % \pgfkeysifdefined{/itf/bookmark/@save/color}%
825      % is quicker than setting a /.style ...
826      % \pgfkeysifdefined{/itf/bookmark/@save/FontStyle}
827      % {\pgfkeysgetvalue{/itf/bookmark/@save/color}\BKM@color}
828      % {}%
829      % \pgfkeysifdefined{/itf/bookmark/@save/grouplevel}
830      % { shortcut: \bookmarksetup{bold*} called inside main group (level 0)
831      % \ifnum\pgfkeysvalueof{/itf/bookmark/@save/grouplevel}=z@
832      %     \global\pgfkeyslet{/itf/bookmark/@save/FontStyle}@\undefined
833      %     \global\pgfkeyslet{/itf/bookmark/@save/color}@\undefined
834      %     \global\pgfkeyslet{/itf/bookmark/@save/grouplevel}@\undefined
835      %     \global\pgfkeyslet{/itf/bookmark/*}@\undefined
836      %     \global\let\BKM@color\BKM@color
837      %     \global\let\BS@BKM@FontStyle\BS@BKM@FontStyle
838      % \else % restore after each group as needed
839      %     \pgfkeyslet{/itf/bookmark/@save/FontStyle}@\undefined
840      %     \pgfkeyslet{/itf/bookmark/@save/color}@\undefined
841      %     \pgfkeyslet{/itf/bookmark/*}@\undefined
842      %     \ifnum\currentgrouplevel>z@
843      %         \aftergroup\itf@BKM@restore
844      %     \else
845      %         \pgfkeyslet{/itf/bookmark/@save/grouplevel}@\undefined
846      % \fi\fi
847      %
848      {}%
849  } ,%
850  @restore/bold/.code={% bold is "hidden" in \BS@BKM@FontStyle
851  \begingroup
852  \edef\itf@tp{\bookmarkget{bold}}%
853  \pgfqkeys{/itf/bookmark/@restore}{FontStyle}%
854  \count@=\bookmarkget{bold}%
855  \edef\itf@tp{\endgroup
856  \ifnum\itf@tp=\count@
857  \else \noexpand\bookmarksetup{%
858  bold\ifcase\count@ =false\fi}\fi
859  }\itf@tp},%
860  @restore/italic/.code={% italic is "hidden" in \BS@BKM@FontStyle
861  \begingroup
862  \edef\itf@tp{\bookmarkget{italic}}%
863  \pgfqkeys{/itf/bookmark/@restore}{FontStyle}%
864  \count@=\bookmarkget{italic}%
865  \edef\itf@tp{\endgroup
866  \ifnum\itf@tp=\count@
867  \else \noexpand\bookmarksetup{%
868  italic\ifcase\count@ =false\fi}\fi
869  }\itf@tp},%
870 }
871 \def\itf@BKM@restore{%
872  \pgfkeysifdefined{/itf/bookmark/@save/grouplevel}
873  {\pgfkeysalso{/itf/bookmark/@restore}}
874  {}%
875 }% \itf@BKM@restore
876 %% Plug the BKM keys in the interfaces pgfkeys (not efficient but quickly written...)
877 \define@key{BKM}{disable}[true]{\itf@pgfkeys\pgfkeysalso{/itf/bookmark/disable={#1}}}
878 \define@key{BKM}{disable*}[true]{\itf@pgfkeys\pgfkeysalso{/itf/bookmark/disable*={#1}}}
879 \define@key{BKM}{bold*}[true]{\itf@pgfkeys\pgfkeysalso{/itf/bookmark:bold*={#1}}}
880 \define@key{BKM}{italic*}[true]{\itf@pgfkeys\pgfkeysalso{/itf/bookmark:italic*={#1}}}
881 \define@key{BKM}{color*}[]{\itf@pgfkeys\pgfkeysalso{/itf/bookmark:color*={#1}}}
882 \define@key{BKM}{atend}[true]{\itf@pgfkeys\pgfkeysalso{/itf/bookmark/atend={#1}}}%
```

2) restore previous values in case starred* keys have been set.

```

883 \def\itf@BKM@write#1{%
884   \aftergroup\itf@BKM@restore
885   \ifcase\itf@BKM@enable
886     \global\let\itf@BKM@enable\@ne
887   \or
888     \immediate\write\@mainaux{\#1}%
889   \else
890     \ifcase -\itf@BKM@enable % (case -0 = 0 tested before)
891     \or % (case -1 = enable but At End)
892       \let\itf@BKM@enable\@ne
893       \edef\itf@tp{\immediate\write\@mainaux{\#1}}%
894       \expandafter\BookmarkAtEnd\expandafter{\itf@tp}%
895     \fi
896   \fi
897 }% \itf@BKM@write

```

### 21.7.2 ref and pageref keys

\itf@BKM@pageref

```

898 %% pageref= key for \bookmark (using refcount : \getpagerefnumber)
899 \def\itf@BKM@pageref[#1]{%
900   \expandafter\itf@BKM@@pageref
901   \csname r@#1\endcsname{#1}%
902 }% \itf@BKM@pageref
903 \def\itf@BKM@@pageref#1#2{%
904   \ifx#1\relax
905     \protect\G@refundefinedtrue
906     \@latex@warning{%
907       Hyper reference '#2' on page \thepage \space undefined
908       \MessageBreak for bookmark destination page
909     }%
910   \fi
911   \edef\BKM@page{\getrefbykeydefault{#2}{page}{1}}%
912 }% \itf@BKM@@pageref
913 \define@key{BKM}{pageref}{%
914   \Hy@babelnormalise\itf@BKM@pageref[{"#1"}]}

```

\itf@BKM@ref

```

915 %% ref= key for \bookmark
916 \def\itf@BKM@ref[#1]{%
917   \expandafter\itf@BKM@@ref
918   \csname r@#1\endcsname{#1}%
919 }% \itf@BKM@ref
920 \def\itf@BKM@@ref#1#2{%
921   \ifx#1\relax
922     \protect\G@refundefinedtrue
923     \@latex@warning{%
924       Hyper reference '#2' on page \thepage \space undefined
925       \MessageBreak for bookmark destination
926     }%
927     \def\BKM@dest{??}%
928   \else
929     \edef\BKM@dest{\expandafter\@fourthoffive#1\empty\empty\empty\empty}%
930   \fi
931 }% \BKM@@ref
932 \define@key{BKM}{ref}{%
933   \Hy@babelnormalise\itf@BKM@ref[{"#1"}]}

```

```
934 </bookmark>
```

## 21.8 † Interface hypbmsec

### 21.8.1 interfaces-hypbmsec

```
935 {*hypbmsec}
936 \ProvidesPackage{interfaces-hypbmsec}
937     [2010/11/15 v2.1 lastsection etc. (FC)]
938 \RequirePackage{interfaces-base}
939 \itf@Check@Package{hypbmsec}
940 \AfterPackage{*{interfaces-titlesec}{% requires interface-titlesec
941     \input{interfaces-hypbmsec.code.tex}%
942 }%
943 </hypbmsec>
```

### 21.8.2 interfaces-hypbmsec.code.tex

```
944 {*hypbmsec.code.tex}
945 \ProvidesFile{interfaces-hypbmsec.code.tex}
946     [2010/10/27 v1.2 code for interfaces-hypbmsec (FC)]
```

\hbs@do (patch)

```
947 \itf@patch\pretocmd\hbs@do{%
948     \edef\itf@tp{\expandafter\itf@strip@at\string#1}%
949 }%
950 {\PackageInfo{interfaces-hypbmsec}{%
951     {Command \string\hbs@do\space successfully patched}}}
952 {\PackageWarning{interfaces-hypbmsec}{%
953     {Command \string\hbs@do\space could not be patched
954     \MessageBreak loading of ‘interfaces-hypbmsec’ aborted
955     \MessageBreak \string\lastsection\space will not work}%
956 \endinput}}
```

\hbs@process (patch)

```
957 \itf@patch\pretocmd\hbs@process{%
958     \itf@hbs@next{\#1}%
959 }%
960 {\PackageInfo{interfaces-hypbmsec}{%
961     {Command \string\hbs@process\space successfully patched}}}
962 {\PackageWarning{interfaces-hypbmsec}{%
963     {Command \string\hbs@process\space could not be patched
964     \MessageBreak loading of ‘interfaces-hypbmsec’ aborted
965     \MessageBreak \string\lastsection\space will not work}%
966 \endinput}}%
```

\itf@hbs@next

```
967 \long\def\itf@hbs@next{\#1}%
968 \csdef{\itf@next\itf@tp title}{\#1}%
969 \expandafter\let\csname itf@next\itf@tp toc\expandafter\endcsname
970 \csname
971     \ifx\hbs@tocstring\relax \itf@next\itf@tp title\else
972             hbs@tocstring\fi
973     \endcsname
974 \expandafter\let\csname itf@next\itf@tp bookmark\expandafter\endcsname
975 \csname
976     \ifx\hbs@bmstring\relax \itf@next\itf@tp toc\else
977             hbs@bmstring\fi
978     \endcsname
979 \expandafter\expandafter\expandafter
```

```

980 \expandafter\expandafter\expandafter
981 \expandafter\itf@hbs@chkclass
982     \expandafter\expandafter\expandafter\string
983         \expandafter\hbs@seccmd\string\ttl@\@nil
984     {\expandafter\itf@last@from@next\expandafter{\itf@tp}}%
985 }% \itf@hbs@next
986 \long\expandafter\def\expandafter\itf@hbs@chkclass
987     \expandafter#\expandafter1\string\ttl@#2\@nil{%
988 \ifblank{#1}{\ifnobreak\itf@hbs@nobreakfix#2\@nnil\fi}%
989 }% \itf@hbs@chkclass
990 \def\itf@hbs@nobreakfix#1{\itf@hbs@n@breakfix}
991 \def\itf@hbs@n@breakfix#1#2\@nnil{%
992     \unless\ifx\lastsectionname\relax
993     \unless\ifnum\csname\ttl@#1\endcsname > \csname\ttl@#1\endcsname\relax
994         \nobreakfalse
995     \fi\fi
996 }% \itf@hbs@nobreakfix

\itf@last@from@next

997 \def\itf@last@from@next#1{%
998     \gdef\lastsectionname{#1}%
999     \global\expandafter\let\csname last#1title\expandafter\endcsname
1000         \csname itf@next#1title\endcsname
1001     \global\expandafter\let\csname last#1toc\expandafter\endcsname
1002         \csname itf@next#1toc\endcsname
1003     \global\expandafter\let\csname last#1bookmark\expandafter\endcsname
1004         \csname itf@next#1bookmark\endcsname
1005     \let@elt\itf@resetlastsections
1006     \csname cl@#1\endcsname
1007 }% \itf@last@from@next
1008 \def\itf@resetlastsections#1{%
1009     \expandafter\let\csname last#1title\endcsname\relax
1010     \expandafter\let\csname last#1toc\endcsname\relax
1011     \expandafter\let\csname last#1bookmark\endcsname\relax
1012     \csname cl@#1\endcsname
1013 }% \itf@resetlastsections

1014 \let\lastsectionname\relax
1015 @for\itf@tp:=part,section,subsection,subsubsection,paragraph,subparagraph%
1016     \do{\expandafter\let\csname last\itf@tp title\endcsname\relax
1017         \expandafter\let\csname last\itf@tp toc\endcsname\relax
1018         \expandafter\let\csname last\itf@tp bookmark\endcsname\relax
1019     }%
1020 \AtBeginDocument{\ifundefined\chapter{}%
1021     {\let\lastchaptertitle\relax
1022     \let\lastchaptertos\relax
1023     \let\lastchapterbookmark\relax}%
1024 }%
1025 
```

## 21.9 + Interface fancyhdr : \pagestyle

```
1026 {*fancyhdr}
1027 \ProvidesPackage{interfaces-fancyhdr}
1028   [2010/11/15 v2.1 pgfkeys macros for fancyhdr (FC)]
1029 \RequirePackage{interfaces-base}
1030 \itf@Check@Package{fancyhdr}
1031 \itf@pgfkeys
1032 \robustify\fancyhf
1033 \robustify\fancyhffoffset
```

### 21.9.1 \pagestyle

\pagestyle

```
1034 \newrobustcmd*\pagestyle{%
1035   \let\fancy@gb\relax % Just in case...
1036   \@ifstar
1037     {\itf@pagestyle{}}
1038     {\itf@pagestyle{reset,}{}}
1039 }% \pagestyle
1040 \def\itf@pagestyle#1{\@testopt
1041   {\itf@p@gesetup{#1}}{}%
1042 }% \itf@pagestyle
1043 \def\itf@p@gesetup#1[#2]#3{\ifblank{#2}
1044   {% modifies the current page style
1045     \itf@pgfkeys\pgfqkeys{/itf/fancyhdr}{#1#3}\relax\ps@@fancy}%
1046   {% defines or modifies a given page style
1047     \ifblank{#1}
1048       \csappto% star form: keep existing value (if not overwritten)
1049       \csdef% no star form: overwrite (+ reset at \pagestyle time)
1050       {ps@#2}{\itf@pgfkeys\pgfqkeys{/itf/fancyhdr}{#1#3}\relax\ps@@fancy}%
1051   }%
1052 }% \itf@p@gesetup
1053 \pgfqkeys{/itf/pagestyle}{%
1054   @current/.is private=interfaces-fancyhdr,%
1055   @style/.is private=interfaces-fancyhdr,%
1056 }
```

\pagestyle

```
1057 \renewrobustcmd*\pagestyle[1]{\ifcsundef{ps@#1}
1058   {\itf@undefinedpagestyle{#1}}
1059   {\csname ps@#1\endcsname}%
1060 }% \pagestyle
```

\thispagestyle

```
1061 \renewrobustcmd*\thispagestyle[1]{\ifcsundef{ps@#1}
1062   {\itf@undefinedpagestyle{#1}}
1063   {\global\@specialpagetrue\gdef\@specialstyle{#1}}%
1064 }% \thispagestyle
1065 \newcommand*\itf@undefinedpagestyle[1]{%
1066   \@latex@error{Undefined page style: #1
1067   \MessageBreak your command will be ignored}\@eha
1068 }% \itf@undefinedpagestyle
```

## 21.9.2 Keys for \pagesetup

```

1069 \pgfkeys{%
1070   /handlers/.fancy@hf/.code=%
1071   \edef\itf@tp{\pgfkeyscurrentpath}%
1072   \pgfkeysalso{\pgfkeyscurrentpath/.store as code=}%
1073   \pgfkeysalso{\pgfkeyscurrentpath/.append code/.expanded=%
1074     \csdef{f@ncy#1}{%
1075       \noexpand\pgfkeysvalueof{\itf@tp/font}%
1076       \noexpand\pgfkeysvalueof{\itf@tp}\noexpand\strut},%
1077 }% handler for fancyhdr interface
1078 \pgfqkeys{/itf/fancyhdr}{% keys for \pagesetup
1079   code/.code={#1},% execution at expansion of \ps@... => \pgfqkeys{...,code={...},...}
1080   page style/.style={code=\pagestyle{#1}},%
1081   page style/.alias=pagestyle,%
1082   load style/.alias=pagestyle,%
1083   head rule/.store as dimen in cs=headrulewidth,%
1084   head rule/.default=.4pt,%
1085   head rule/.blank=\z@,%
1086     head rule/.alias*=head rule,%
1087     headrulewidth/.alias*=head rule,%
1088     head rule/width/.alias*=head rule,%
1089   head rule/font/.store as code,%
1090   head rule/color/.style={head rule/font+=\color{#1}},%
1091   headruler/.code=%
1092     \let\headrule\pgfkeyscurrentvalue
1093     \ifx\headrule\@empty\else
1094       \preto\headrule{\bgroup\pgfkeysvalueof{/itf/fancyhdr/head rule/font}}%
1095       \appto\headrule{\egroup}%
1096     \fi,%
1097     headruler/.default=%
1098       \if@fancyplain\let\headrulewidth\plainheadrulewidth\fi
1099       \hrule@height\headrulewidth \@width\headwidth
1100       \vskip-\headrulewidth\relax,%
1101     headrule/.alias=headruler,%
1102     headrule,% sets default
1103   foot rule/.store as dimen in cs=footrulewidth,%
1104   foot rule/.default=.4pt,%
1105   foot rule/.blank=\z@,%
1106   foot rule/width/.alias*=foot rule,%
1107   foot rule/width/.alias*=foot rule,%
1108   foot rule/width/.alias*=foot rule,%
1109   foot rule/font/.store as code,%
1110   foot rule/color/.style={foot rule/font+=\color{#1}},%
1111   footruler/.code=%
1112     \let\footrule\pgfkeyscurrentvalue
1113     \ifx\footrule\@empty\else
1114       \preto\footrule{\bgroup\pgfkeysvalueof{/itf/fancyhdr/head rule/font}}%
1115       \appto\footrule{\egroup}%
1116     \fi,%
1117     footruler/.default=%
1118       \if@fancyplain\let\footrulewidth\plainfootrulewidth\fi
1119       \vskip-2\footrulewidth
1120       \hrule@height\footrulewidth \@width\headwidth\vskip\footrulewidth,%
1121     footrule/.alias=footruler,%
1122     footrule,% sets default
1123   rule/.style={head rule={#1},foot rule={#1}},%
1124   rule+.style={head rule+={#1},foot rule+={#1}},%
1125   rule-.style={head rule-={#1},foot rule-={#1}},%
1126   rule*.style={head rule*={#1},foot rule*={#1}},%
1127     rules/.alias*=rule,rulewidth/.alias*=rule,%
1128     rule/width/.alias*=rule,rules/width/.alias*=rule,%
1129   norule/.style={rule=\z@},%
1130   norule/.value forbidden,%

```

```

1131      norules/.alias=norule,%  

1132      rule/font/.style={head/rule/font={#1},foot/rule/font={#1}},%  

1133      rule/font+/.style={head/rule/font+={#1},foot/rule/font+={#1}},%  

1134          rule/font*/.alias=rule/font+,%  

1135          rules/font/.alias=rule/font,  

1136          rules/font+/.alias=rule/font+,%  

1137          rules/font*/.alias=rule/font+,%  

1138      rule/color/.style={head/rule/color={#1},foot/rule/color={#1}},%  

1139          rules/color/.alias=rule/color,%  

1140      foot/rule/skip/.store as glue in cs=footruleskip,%  

1141          foot/rule/skip/.default=.3\normalbaselineskip,%  

1142          foot/rule/skip/.blank=\z@,%  

1143          foot/ruleskip/.alias*=foot/rule/skip,%           => puis les alias&style: /.alias 2  

1144          footruleskip/.alias*=foot/rule/skip,%           crée aussi  

1145      font/.style={head/font={#1},foot/font={#1}},%  

1146          font+/.style={head/font+={#1},foot/font+={#1}},%  

1147          font*/.alias=font+,%  

1148      color/.style={head/color={#1},foot/color={#1},head/rule/color={#1},foot/rule/color={#1}},%  

1149      head/.style={head/left={#1},head/center={#1},head/right={#1}},%  

1150          head/font/.style={head/left/font={#1},head/center/font={#1},head/right/font={#1}},%  

1151          head/font+/.style={head/left/font+={#1},head/center/font+={#1},head/right/font+={#1}},%  

1152          head/font*/.alias=head/font+,%  

1153          head/color/.style={head/left/color={#1},head/center/color={#1},head/right/color={#1},he  

nohead/.style={head=,head/font=,head/rule=,head/font=},  

1155      foot/.style={foot/left={#1},foot/center={#1},foot/right={#1}},%  

1156          foot/font/.style={foot/left/font={#1},foot/center/font={#1},foot/right/font={#1}},%  

1157          foot/font+/.style={foot/left/font+={#1},foot/center/font+={#1},foot/right/font+={#1}},%  

1158          foot/font*/.alias=foot/font+,%  

1159          foot/color/.style={foot/left/color={#1},foot/center/color={#1},foot/right/color={#1},fo  

nofoot/.style={foot=,foot/font=,foot/rule=,foot/font=},  

1161      left/.style={head/left={#1},foot/left={#1}},%  

1162          left/font/.style={head/left/font={#1},foot/left/font={#1}},%  

1163          left/font+/.style={head/left/font+={#1},foot/left/font+={#1}},%  

1164          left/font*/.alias=left/font+,%  

1165          left/color/.style={head/left/color={#1},foot/left/color={#1}},%  

1166      center/.style={head/center={#1},foot/center={#1}},%  

1167          center/font/.style={head/center/font={#1},foot/center/font={#1}},%  

1168          center/font+/.style={head/center/font+={#1},foot/center/font+={#1}},%  

1169          center/font*/.alias=center/font+,%  

1170          center/color/.style={head/center/color={#1},foot/center/color={#1}},%  

1171      right/.style={head/right={#1},foot/right={#1}},%  

1172          right/font/.style={head/right/font={#1},foot/right/font={#1}},%  

1173          right/font+/.style={head/right/font+={#1},foot/right/font+={#1}},%  

1174          right/font*/.alias=right/font+,%  

1175          right/color/.style={head/right/color={#1},foot/right/color={#1}},%  

1176      head/left/.style={head/left/even={#1},head/left/odd={#1}},%  

1177          head/left/font/.style={head/left/even/font={#1},head/left/odd/font={#1}},%  

1178          head/left/font+/.style={head/left/even/font+={#1},head/left/odd/font+={#1}},%  

1179          head/left/font*/.alias=head/left/font+,%  

1180          head/left/color/.style={head/left/even/color={#1},head/left/odd/color={#1}},%  

1181      head/center/.style={head/center/even={#1},head/center/odd={#1}},%  

1182          head/center/font/.style={head/center/even/font={#1},head/center/odd/font={#1}},%  

1183          head/center/font+/.style={head/center/even/font+={#1},head/center/odd/font+={#1}},%  

1184          head/center/font*/.alias=head/center/font+,%  

1185          head/center/color/.style={head/center/even/color={#1},head/center/odd/color={#1}},%  

1186      head/right/.style={head/right/even={#1},head/right/odd={#1}},%  

1187          head/right/font/.style={head/right/even/font={#1},head/right/odd/font={#1}},%  

1188          head/right/font+/.style={head/right/even/font+={#1},head/right/odd/font+={#1}},%  

1189          head/right/font*/.alias=head/right/font+,%  

1190          head/right/color/.style={head/right/even/color={#1},head/right/odd/color={#1}},%  

1191      foot/left/.style={foot/left/even={#1},foot/left/odd={#1}},%  

1192          foot/left/font/.style={foot/left/even/font={#1},foot/left/odd/font={#1}},%  

1193          foot/left/font+/.style={foot/left/even/font+={#1},foot/left/odd/font+={#1}},%  

1194          foot/left/font*/.alias=foot/left/font+,%  

1195          foot/left/color/.style={foot/left/even/color={#1},foot/left/odd/color={#1}},%

```

```

1196   foot/center/.style={foot/center/even={#1},foot/center/odd={#1}},%
1197     foot/center/font/.style={foot/center/even/font={#1},foot/center/odd/font={#1}},%
1198     foot/center/font+/.style={foot/center/even/font+={#1},foot/center/odd/font+={#1}},%
1199     foot/center/font*/.alias=foot/center/font+,%
1200     foot/center/color/.style={foot/center/even/color={#1},foot/center/odd/color={#1}},%
1201   foot/right/.style={foot/right/even={#1},foot/right/odd={#1}},%
1202     foot/right/font/.style={foot/right/even/font={#1},foot/right/odd/font={#1}},%
1203     foot/right/font+/.style={foot/right/even/font+={#1},foot/right/odd/font+={#1}},%
1204     foot/right/font*/.alias=foot/right/font+,%
1205     foot/right/color/.style={foot/right/even/color={#1},foot/right/odd/color={#1}},%
1206   head/left/odd/.fancy@hf=olh,%
1207   head/left/odd/font/.store as code,%
1208   head/left/odd/color/.style={head/left/odd/font+=\color{#1}},%
1209     odd/head/left/.alias=head/left/odd,%
1210     odd/head/left/font/.alias*=head/left/odd/font,%
1211     odd/head/left/color/.alias=head/left/odd/color,%
1212     head/odd/left/.alias=head/left/odd,%
1213     head/odd/left/font/.alias*=head/left/odd/font,%
1214     head/odd/left/color/.alias=head/left/odd/color,%
1215   head/center/odd/.fancy@hf=och,%
1216   head/center/odd/font/.store as code,%
1217   head/center/odd/color/.style={head/center/odd/font+=\color{#1}},%
1218     odd/head/center/.alias=head/center/odd,%
1219     odd/head/center/font/.alias*=head/center/odd/font,%
1220     odd/head/center/color/.alias=head/center/odd/color,%
1221     head/odd/center/.alias=head/center/odd,%
1222     head/odd/center/font/.alias*=head/center/odd/font,%
1223     head/odd/center/color/.alias=head/center/odd/color,%
1224   head/right/odd/.fancy@hf=orh,%
1225   head/right/odd/font/.store as code,%
1226   head/right/odd/color/.style={head/right/odd/font+=\color{#1}},%
1227     odd/head/right/.alias=head/right/odd,%
1228     odd/head/right/font/.alias*=head/right/odd/font,%
1229     odd/head/right/color/.alias=head/right/odd/color,%
1230     head/odd/right/.alias=head/right/odd,%
1231     head/odd/right/font/.alias*=head/right/odd/font,%
1232     head/odd/right/color/.alias=head/right/odd/color,%
1233   head/left/even/.fancy@hf=elh,%
1234   head/left/even/font/.store as code,%
1235   head/left/even/color/.style={head/left/even/font+=\color{#1}},%
1236     even/head/left/.alias=head/left/even,%
1237     even/head/left/font/.alias*=head/left/even/font,%
1238     even/head/left/color/.alias=head/left/even/color,%
1239     head/even/left/.alias=head/left/even,%
1240     head/even/left/font/.alias*=head/left/even/font,%
1241     head/even/left/color/.alias=head/left/even/color,%
1242   head/center/even/.fancy@hf=ech,%
1243   head/center/even/font/.store as code,%
1244   head/center/even/color/.style={head/center/even/font+=\color{#1}},%
1245     even/head/center/.alias=head/center/even,%
1246     even/head/center/font/.alias*=head/center/even/font,%
1247     even/head/center/color/.alias=head/center/even/color,%
1248     head/even/center/.alias=head/center/even,%
1249     head/even/center/font/.alias*=head/center/even/font,%
1250     head/even/center/color/.alias=head/center/even/color,%
1251   head/right/even/.fancy@hf=erh,%
1252   head/right/even/font/.store as code,%
1253   head/right/even/color/.style={head/right/even/font+=\color{#1}},%
1254     even/head/right/.alias=head/right/even,%
1255     even/head/right/font/.alias*=head/right/even/font,%
1256     even/head/right/color/.alias=head/right/even/color,%
1257     head/even/right/.alias=head/right/even,%
1258     even/head/right/font/.alias*=head/right/even/font,%
1259     even/head/right/color/.alias=head/right/even/color,%
1260   foot/left/odd/.fancy@hf=olf,%

```

```

1261 foot/left/odd/font/.store as code,%
1262 foot/left/odd/color/.style={foot/left/odd/font+=\color{#1}},%
1263     odd/foot/left/.alias=foot/left/odd,%
1264     odd/foot/left/font/.alias*=foot/left/odd/font,%
1265     odd/foot/left/color/.alias=foot/left/odd/color,%
1266     foot/odd/left/.alias=foot/left/odd,%
1267     foot/odd/left/font/.alias*=foot/left/odd/font,%
1268     foot/odd/left/color/.alias=foot/left/odd/color,%
1269 foot/center/odd/.fancy@hf=ocf,%
1270 foot/center/odd/font/.store as code,%
1271 foot/center/odd/color/.style={foot/center/odd/font+=\color{#1}},%
1272     odd/foot/center/.alias=foot/center/odd,%
1273     odd/foot/center/font/.alias*=foot/center/odd/font,%
1274     odd/foot/center/color/.alias=foot/center/odd/color,%
1275     foot/odd/center/.alias=foot/center/odd,%
1276     foot/odd/center/font/.alias*=foot/center/odd/font,%
1277     foot/odd/center/color/.alias=foot/center/odd/color,%
1278 foot/right/odd/.fancy@hf=orf,%
1279 foot/right/odd/font/.store as code,%
1280 foot/right/odd/color/.style={foot/right/odd/font+=\color{#1}},%
1281     odd/foot/right/.alias=foot/right/odd,%
1282     odd/foot/right/font/.alias*=foot/right/odd/font,%
1283     odd/foot/right/color/.alias=foot/right/odd/color,%
1284     odd/foot/right/.alias=foot/right/odd,%
1285     odd/foot/right/font/.alias*=foot/right/odd/font,%
1286     odd/foot/right/color/.alias=foot/right/odd/color,%
1287 foot/left/even/.fancy@hf=elf,%
1288 foot/left/even/font/.store as code,%
1289 foot/left/even/color/.style={foot/left/even/font+=\color{#1}},%
1290     even/foot/left/.alias=foot/left/even,%
1291     even/foot/left/font/.alias*=foot/left/even/font,%
1292     even/foot/left/color/.alias=foot/left/even/color,%
1293     foot/even/left/.alias=foot/left/even,%
1294     foot/even/left/font/.alias*=foot/left/even/font,%
1295     foot/even/left/color/.alias=foot/left/even/color,%
1296 foot/center/even/.fancy@hf=ecf,%
1297 foot/center/even/font/.store as code,%
1298 foot/center/even/color/.style={foot/center/even/font+=\color{#1}},%
1299     even/foot/center/.alias=foot/center/even,%
1300     even/foot/center/font/.alias*=foot/center/even/font,%
1301     even/foot/center/color/.alias=foot/center/even/color,%
1302     foot/even/center/.alias=foot/center/even,%
1303     foot/even/center/font/.alias*=foot/center/even/font,%
1304     foot/even/center/color/.alias=foot/center/even/color,%
1305 foot/right/even/.fancy@hf=erf,%
1306 foot/right/even/font/.store as code,%
1307 foot/right/even/color/.style={foot/right/even/font+=\color{#1}},%
1308     even/foot/right/.alias=foot/right/even,%
1309     even/foot/right/font/.alias*=foot/right/even/font,%
1310     even/foot/right/color/.alias=foot/right/even/color,%
1311     foot/even/right/.alias=foot/right/even,%
1312     foot/even/right/font/.alias*=foot/right/even/font,%
1313     foot/even/right/color/.alias=foot/right/even/color,%
1314 offset/.style={head/offset={#1},foot/offset={#1}},%
1315     offset+/.style={head/offset+={#1},foot/offset+={#1}},%
1316     offset-/.style={head/offset-={#1},foot/offset-={#1}},%
1317     offset*/.style={head/offset*={#1},foot/offset*={#1}},%
1318 head/offset/.style={head/left/offset={#1},head/right/offset={#1}},%
1319     head/offset+/.style={head/left/offset+={#1},head/right/offset+={#1}},%
1320     head/offset-/.style={head/left/offset-={#1},head/right/offset-={#1}},%
1321     head/offset*/.style={head/left/offset*={#1},head/right/offset*={#1}},%
1322     offset/head/.alias*=head/offset,%
1323 foot/offset/.style={foot/left/offset={#1},foot/right/offset={#1}},%
1324     foot/offset+/.style={foot/left/offset+={#1},foot/right/offset+={#1}},%
1325     foot/offset-/.style={foot/left/offset-={#1},foot/right/offset-={#1}},%

```

```

1326     foot/offset*/.style={foot/left/offset*={#1},foot/right/offset*={#1}},%
1327         offset/foot/.alias*=foot/offset,%
1328     left/offset/.style={foot/left/offset={#1},head/left/offset={#1}},%
1329         left/offset+/.style={foot/left/offset+={#1},head/left/offset+={#1}},%
1330         left/offset-/.style={foot/left/offset-={#1},head/left/offset-={#1}},%
1331         left/offset*/.style={foot/left/offset*={#1},head/left/offset*={#1}},%
1332             offset/left/.alias*=left/offset,%
1333     right/offset/.style={foot/right/offset={#1},head/right/offset={#1}},%
1334         right/offset+/.style={foot/right/offset+={#1},head/right/offset+={#1}},%
1335         right/offset-/.style={foot/right/offset-={#1},head/right/offset-={#1}},%
1336         right/offset*/.style={foot/right/offset*={#1},head/right/offset*={#1}},%
1337             offset/right/.alias*=right/offset,%
1338     head/left/offset/.style={head/left/even/offset={#1},head/left/odd/offset={#1}},%
1339         head/left/offset+/.style={head/left/even/offset+={#1},head/left/odd/offset+={#1}},%
1340         head/left/offset-/.style={head/left/even/offset-={#1},head/left/odd/offset-={#1}},%
1341         head/left/offset*/.style={head/left/even/offset*={#1},head/left/odd/offset*={#1}},%
1342             offset/head/left/.alias*=head/left/offset,%
1343             offset/left/head/.alias*=head/left/offset,%
1344             left/head/offset/.alias*=head/left/offset,%
1345     head/right/offset/.style={head/right/even/offset={#1},head/right/odd/offset={#1}},%
1346         head/right/offset+/.style={head/right/even/offset+={#1},head/right/odd/offset+={#1}},%
1347         head/right/offset-/.style={head/right/even/offset-={#1},head/right/odd/offset-={#1}},%
1348         head/right/offset*/.style={head/right/even/offset*={#1},head/right/odd/offset*={#1}},%
1349             offset/head/right/.alias*=head/right/offset,%
1350             offset/right/head/.alias*=head/right/offset,%
1351             right/head/offset/.alias*=head/right/offset,%
1352     foot/left/offset/.style={foot/left/even/offset={#1},foot/left/odd/offset={#1}},%
1353         foot/left/offset+/.style={foot/left/even/offset+={#1},foot/left/odd/offset+={#1}},%
1354         foot/left/offset-/.style={foot/left/even/offset-={#1},foot/left/odd/offset-={#1}},%
1355         foot/left/offset*/.style={foot/left/even/offset*={#1},foot/left/odd/offset*={#1}},%
1356             offset/foot/left/.alias*=foot/left/offset,%
1357             offset/left/foot/.alias*=foot/left/offset,%
1358             left/foot/offset/.alias*=foot/left/offset,%
1359     foot/right/offset/.style={foot/right/even/offset={#1},foot/right/odd/offset={#1}},%
1360         foot/right/offset+/.style={foot/right/even/offset+={#1},foot/right/odd/offset+={#1}},%
1361         foot/right/offset-/.style={foot/right/even/offset-={#1},foot/right/odd/offset-={#1}},%
1362         foot/right/offset*/.style={foot/right/even/offset*={#1},foot/right/odd/offset*={#1}},%
1363             offset/foot/right/.alias*=foot/right/offset,%
1364             offset/right/foot/.alias*=foot/right/offset,%
1365             right/foot/offset/.alias*=foot/right/offset,%
1366     head/left/even/offset/.store in skip cs=f@ncy0@elh,%
1367         head/left/even/offset/.append code=\fancy@setoffs,%
1368         offset/head/left/even/.alias*=head/left/even/offset,%
1369         left/even/head/offset/.alias*=head/left/even/offset,%
1370     head/left/odd/offset/.store in skip cs=f@ncy0@olh,%
1371         head/left/odd/offset/.append code=\fancy@setoffs,%
1372         offset/head/left/odd/.alias*=head/left/odd/offset,%
1373         offset/head/left/odd/.alias*=head/left/odd/offset,%
1374         left/odd/head/offset/.alias*=head/left/odd/offset,%
1375     head/right/even/offset/.store in skip cs=f@ncy0@erh,%
1376         head/right/even/offset/.append code=\fancy@setoffs,%
1377         offset/head/right/even/.alias*=head/right/even/offset,%
1378         right/even/head/offset/.alias*=head/right/even/offset,%
1379     head/right/odd/offset/.store in skip cs=f@ncy0@orh,%
1380         head/right/odd/offset/.append code=\fancy@setoffs,%
1381         offset/head/right/odd/.alias*=head/right/odd/offset,%
1382         right/odd/head/offset/.alias*=head/right/odd/offset,%
1383     foot/left/even/offset/.store in skip cs=f@ncy0@elf,%
1384         foot/left/even/offset/.append code=\fancy@setoffs,%
1385         offset/foot/left/even/.alias*=foot/left/even/offset,%
1386         left/even/foot/offset/.alias*=foot/left/even/offset,%
1387     foot/left/odd/offset/.store in skip cs=f@ncy0@olf,%
1388         foot/left/odd/offset/.append code=\fancy@setoffs,%
1389         offset/foot/left/odd/.alias*=foot/left/odd/offset,%
1390         left/odd/foot/offset/.alias*=foot/left/odd/offset,%

```

```

1391   foot/right/even/offset/.store in skip cs=f@ncy0@erf,%
1392     foot/right/even/offset/.append code=\fancy@setoffs,%
1393       offset/foot/right/even/.alias*=foot/right/even/offset,%
1394         right/even/foot/offset/.alias*=foot/right/even/offset,%
1395   foot/right/odd/offset/.store in skip cs=f@ncy0@orf,%
1396     foot/right/odd/offset/.append code=\fancy@setoffs,%
1397       offset/foot/right/odd/.alias*=foot/right/odd/offset,%
1398         right/odd/foot/offset/.alias*=foot/right/odd/offset,%
1399   reset/.style={norule,head=,foot=,offset=\z@,foot/rule/skip,rule/font=,font=},%
1400     reset/.value forbidden,%
1401 }% keys for \pagesetup (stored into token \itf@tk)
1402 \AtBeginDocument{%
1403 \ifdim\headwidth<\z@
1404 %% This catches the case that \headwidth hasn't been initialized and the
1405 %% case that the user added something to \headwidth in the expectation that
1406 %% it was initialized to \textwidth. We compensate this now. This loses if
1407 %% the user intended to multiply it by a factor. But that case is more
1408 %% likely done by saying something like \headwidth=1.2\textwidth.
1409 %% The doc says you have to change \headwidth after the first call to
1410 %% \pagestyle{fancy}. This code is just to catch the most common cases were
1411 %% that requirement is violated.
1412   \global\advance\headwidth123456789sp\global\advance\headwidth\textwidth
1413 \fi}
1414 
```

## 21.10 marks

```

1415 <*marks>
1416 \ProvidesPackage{interfaces-marks}
1417   [2010/10/27 v1.2 easier use of e-TeX marks (FC)]
1418 \RequirePackage{interfaces-base}
```

\marksthe

```

1419 \newrobustcmd\marksthe[2]{%
1420   \ifcsname marks@\#1\endcsname\else
1421     \itf@marks@new{\#1}%
1422   \fi
1423   \expandafter\marks\csname marks@\#1\endcsname{\unexpanded{\#2}}%
1424 }% \marksthe
```

\marksthecs

```

1425 \newrobustcmd*\marksthecs[2]{%
1426   \ifcsname marks@\#1\endcsname\else
1427     \itf@marks@new{\#1}%
1428   \fi
1429   \ifcsundef{\#2}
1430     {}
1431     {\expandafter\marks\csname marks@\#1\endcsname{%
1432       \unexpanded\expandafter\expandafter\expandafter\expandafter{%
1433         \csname\#2\endcsname}}}%
1434 }% \marksthecs
```

\itf@marks@new

```

1435 \def\itf@marks@new#1{%
1436   \expandafter\newmarks\csname marks@\#1\endcsname
1437   \PackageInfo{interfaces-marks}{%
1438     {New marks '#1' defined}}
1439 }% \itf@marks@warning
```

```

\thefirstmarks
\thebotmarks
\thetopmarks
1440 \newcommand*\thefirstmarks[1]{%
1441   \ifcsname marks@\#1\endcsname
1442     \expandafter\firstmarks\csname marks@\#1\endcsname
1443   \fi
1444 }% \thefirstmarks
1445 \newcommand*\thebotmarks[1]{%
1446   \ifcsname marks@\#1\endcsname
1447     \expandafter\botmarks\csname marks@\#1\endcsname
1448   \fi
1449 }% \thebotmarks
1450 \newcommand*\thetopmarks[1]{%
1451   \ifcsname marks@\#1\endcsname
1452     \expandafter\topmarks\csname marks@\#1\endcsname
1453   \fi
1454 }% \thetopmarks

\getthemarks
1455 %% \getthemarks\firstmarks{section}\macro
1456 \newrobustcmd*\getthemarks[3]{%
1457   \ifcsname marks@\#2\endcsname
1458     \expandafter\itf@getthemarks\expandafter#1\csname marks@\#2\endcsname{#3}%
1459   \else \let#3@\undefined
1460   \fi
1461 }% \getthemarks
1462 \def\itf@getthemarks#1#2#3{%
1463   \expandafter\def\expandafter#3\expandafter{\#1#2}%
1464 }% \itf@getthemarks

\getthefirstmarks
\getthebotmarks
\getthetopmarks
1465 \newrobustcmd*\getthefirstmarks[2]{\getthemarks\firstmarks{\#1}{\#2}}
1466 \newrobustcmd*\getthebotmarks[2]{\getthemarks\botmarks{\#1}{\#2}}
1467 \newrobustcmd*\getthetopmarks[2]{\getthemarks\topmarks{\#1}{\#2}}


\ifmarksequal
1468 %% \ifmarksequal\firstmarks\botmarks{subsection}{ true }{ false }
1469 \newrobustcmd*\ifmarksequal[3]{%
1470   \begin{group}
1471     \getthemarks{\#1}{\#3}@tempa
1472     \getthemarks{\#2}{\#3}@tempb
1473   \expandafter\endgroup
1474   \csname ltx@\ifdefined@tempa\ifdefined@tempb
1475     \ifx@\tempa@\tempb first\else
1476       second\fi\else
1477       second\fi\else
1478       second\fi
1479   \oftwo\endcsname
1480 }% \ifmarksequal

1481 </marks>

```

## 21.11 † Interface tocloft : \tocsetup, \lofsetup and \lotsetup

```

1482< *tocloft>
1483 \ProvidesPackage{interfaces-tocloft}
1484   [2010/11/16 v2.2 pgfkeys macros for tocloft (FC)]
1485 \RequirePackage{interfaces-base}
1486 \itf@Check@Package{tocloft}
1487 \itf@pgfkeys

\tocsetup The user commands.
\lofsetup
\lotsetup
\listofsetup
1488 \newrobustcmd\tocsetup[1]{%
1489   \AfterPreamble{\itf@pgfkeys\pgfqkeys{/itf/tocloft/toc}{#1}}%
1490 }% \tocsetup
1491 \newrobustcmd\lofsetup[1]{%
1492   \AfterPreamble{\itf@pgfkeys\pgfqkeys{/itf/tocloft/lof}{#1}}%
1493 }% \lofsetup
1494 \newrobustcmd\lotsetup[1]{%
1495   \AfterPreamble{\itf@pgfkeys\pgfqkeys{/itf/tocloft/lot}{#1}}%
1496 }% \lotsetup
1497 \newrobustcmd\listofsetup[2]{%
1498   \AfterPreamble{\itf@pgfkeys\pgfqkeys{/itf/tocloft/#1}{#2}}%
1499 }% \listofsetup

```

\itf@cft@newlistof This macro creates the keys for toc, lof and lot, and is able to create the keys for each new list created with tocloft's \newlistof.

```

1500 \def\itf@cft@newlistof#1{%
1501   \pgfqkeys{/itf/tocloft/#1}{%
1502     pagestyle/.code=\tocloftpagestyle{##1},%
1503     parskip/.store in skip cs=cftparskip,%
1504       parsep/.alias=parskip,%
1505       mark/.store in cs=cftmark#1,%
1506         marks/.alias=mark,%
1507       label/.store as code,%
1508       ref/.store as code,%
1509         ref/.initial=\csname cft#1title\endcsname,% overwritten (toc,lof,lot)
1510         ref/.default=\csname cft#1title\endcsname,% overwritten (toc,lof,lot)
1511         ref/.blank=\pgfkeysnovalue,%
1512         title/.store as code in cs=cft#1title,% overwritten below for toc, lof and lot
1513         title/top/.store in skip cs=cftbefore#1titleskip,%
1514           beforetitleskip/.alias*=title/top,%
1515           title/bottom/.store in skip cs=cftafter#1titleskip,%
1516             aftertitleskip/.alias*=title/bottom,%
1517             title/font/.store as code in cs=cft#1titlefont,%
1518               title/format/.alias*=title/font,%
1519               title/after/.store as code in cs=cftafter#1title,%
1520               title/command/.code=\csdef{@cftmake#1title}###1{##1},%
1521               dotsep/.code=\itf@units@mu\itf@musk{##1}% converts to mu
1522                 \edef\cftdotsep{\itf@strip@mu\itf@musk},%
1523                 dotsep/.default=4.5mu,%
1524                 dotsep/.blank=\cftnodots,%
1525                 dot/.store in=\cftdot,%
1526                 before/.store as code,%
1527                 after/.store as code,%
1528                 columns/before@hook/.is private=interfaces-tocloft,%
1529                 columns/after@hook/.is private=interfaces-tocloft,%
1530                 columns/before@hook/.initial=,%
1531                 columns/after@hook/.initial=,%
1532                 columns/.code={\ltx@ifpackageloaded{multicol}{%
1533                   \pgfkeysEsetvalue\pgfkeyscurrentkey{\number##1}%
1534                   \ifnum\number##1>\@ne
1535                     \pgfkeysgetvalue{/itf/tocloft/#1/columns/before@hook}\itf@tp
1536                     \ifx\itf@tp@\empty
1537                       \pgfkeyssetvalue{/itf/tocloft/#1/columns/before@hook}{%

```

```

1538     \begingroup
1539     \multicolssep\pgfkeysvalueof{/itf/tocloft/#1/multicolssep}\relax
1540     \begin{multicols}{\pgfkeysvalueof{/itf/tocloft/#1/columns}}%
1541     \columnseprule\pgfkeysvalueof{/itf/tocloft/#1/columns/rule}\relax
1542     \columnsep\pgfkeysvalueof{/itf/tocloft/#1/columns/sep}\relax
1543     \pgfkeysgetvalue{/itf/tocloft/#1/columns/rule/color}\columnseprulecolor
1544     \pgfkeysvalueof{/itf/tocloft/#1/multicols/begin}%
1545   }%
1546   \pgfkeyssetvalue{/itf/tocloft/#1/columns/after@hook}{%
1547     \pgfkeysvalueof{/itf/tocloft/#1/multicols/end}%
1548   \end{multicols}
1549   \endgroup
1550 }%
1551 \fi
1552 \else
1553   \pgfkeyslet\pgfkeyscurrentkey@\ne
1554   \pgfkeyslet{/itf/tocloft/#1/columns/before@hook}@\empty
1555   \pgfkeyslet{/itf/tocloft/#1/columns/after@hook}@\empty
1556 \fi}
1557 {\PackageError{interfaces-tocloft}
1558   {Key ‘columns’: package multicol is required}\@eha}%
1559 },%
1560 columns/.default=2,columns/.blank=1,%
1561   multicols/.alias=columns,%
1562   multicols/.alias=columns,%
1563   twocolumns/.true=\pgfkeysalso{columns=2},%
1564   twocolumns/.false=\pgfkeysalso{columns=1},%
1565   columns/rule/.store as dimen,%
1566   columns/rule/.default=\columnseprule,%
1567   columns/rule,% assign default
1568     columnseprule/.alias*=columns/rule,%
1569     columns/rule/width/.alias*=columns/rule,%
1570     columns/sep/.store as dimen,%
1571     columns/sep/.default=\columnsep,%
1572     columns/sep/.blank=10\p@,%
1573     columns/sep,% assign default
1574       columnsep/.alias*=columns/sep,%
1575     columns/rule/color/.store as color command,%
1576     columns/rule/color,% assign default (\normalcolor)
1577       columnseprulecolor/.alias*=columns/rule/color,%
1578       columns/rulecolor/.alias*=columns/rule/color,%
1579   multicolssep/.store as glue,%
1580     multicolssep/.default=0pt plus4pt minus3pt,%
1581     multicolssep,% sets default
1582   multicols/begin/.initial=,multicols/begin/.default=,%
1583     multicols/before/.alias=multicols/begin,%
1584   multicols/end/.initial=,multicols/end/.default=,%
1585     multicols/after/.alias=multicols/end,%
1586 }% \pgfkeys
1587 \itf@AfterPackage{interfaces-bookmark}{%
1588 \pgfqkeys{/itf/tocloft/#1}{%
1589   bookmark/.is multiple choice,%
1590     bookmark/bold/.true=\pgfkeysaddvalue{/itf/tocloft/#1/bookmark}{}{,bold=true},%
1591     bookmark/bold/.false=\pgfkeysaddvalue{/itf/tocloft/#1/bookmark}{}{,bold=false},%
1592     bookmark/italic/.true=\pgfkeysaddvalue{/itf/tocloft/#1/bookmark}{}{,italic=true},%
1593     bookmark/italic/.false=\pgfkeysaddvalue{/itf/tocloft/#1/bookmark}{}{,italic=false},%
1594     bookmark/color/.code=\pgfkeysaddvalue{/itf/tocloft/#1/bookmark}{}{,color={##1}},%
1595     bookmark/level/.code=\pgfkeysEaddvalue{/itf/tocloft/#1/bookmark}{}{level=%
1596       \ifcsname toclevel##1\endcsname
1597         \csname toclevel##1\endcsname
1598       \else##1\fi},%
1599     bookmark/text/.store as code,%
1600       bookmark/text/.default=\contentsname,%
1601     bookmark/.unknown/.code=\pgfkeysEaddvalue{/itf/tocloft/#1/bookmark}{}{\pgfkeyscurr
1602     bookmark/.initial=,%

```

```

1603  }% \pgfkeys
1604  }% AfterPackage interfaces-bookmark
1605  %% while making a format \jobname is expanded to the name of the format... Therefore, the
1606  \AtEndPreamble{%
1607    \BeforeFile{\jobname.#1}{%
1608      \pgfkeysvalueof{/itf/tocloft/#1/before}%
1609      \pgfkeysvalueof{/itf/tocloft/#1/columns/before@hook}%
1610    }%
1611    \AfterFile{\jobname.#1}{%
1612      \pgfkeysvalueof{/itf/tocloft/#1/columns/after@hook}%
1613      \pgfkeysvalueof{/itf/tocloft/#1/after}%
1614    }%
1615  }%
1616  \ifcsname #1setup\endcsname\else
1617    \pgfqkeys{/itf/tocloft/#1}{%
1618      pagenumbers/.true=\@cftpnumon{#1},%
1619      pagenumbers/.false=\@cftpnumoff{#1},%
1620      pagenumber/.alias=pagenumbers
1621    }%
1622    \expandafter\pretocmd\csname@cftmake#1title\expandafter\endcsname
1623      \expandafter{\expandafter\itf@tocloft\csname cft#1title\endcsname{#1}{listof#1}}
1624    {\PackageInfo{interfaces-tocloft}
1625      {Command \expandafter\string\csname@cftmake#1title\endcsname
1626       successfully augmented}}
1627    {\PackageWarning{interfaces-tocloft}
1628      {Command \expandafter\string\csname@cftmake#1title\endcsname
1629       space could not be patched
1630       \MessageBreak Keys label and bookmark won't work for \string\listofsetup}}%
1631  \fi
1632 }% \itf@cft@newlistof

```

Then create the keys for `toc`, `lof`, and `lot`:

```

1633 \itf@cft@newlistof{toc}
1634 \itf@cft@newlistof{lof}
1635 \itf@cft@newlistof{lot}

1636 %% Specific to toc only -----
1637 \pgfqkeys{/itf/tocloft/toc}{%
1638   depth/.store in counter cs=c@tocdepth,%
1639   depth/.append code=
1640     \global\c@tocdepth\c@tocdepth
1641     \addtocontents{toc}{\c@tocdepth\the\c@tocdepth\relax},%
1642   tocdepth/.alias=depth,%
1643   title/.store as code in cs=contentsname,%
1644   contentsname/.alias=title,%
1645   ref/.initial=\contentsname,%
1646   ref/.default=\contentsname,%
1647   pagenumbers/.style={% boolean key
1648     part/pagenumbers={#1}, chapter/pagenumbers={#1}, section/pagenumbers={#1}, %
1649     subsubsection/pagenumbers={#1}, subsubsubsection/pagenumbers={#1}, %
1650     paragraph/pagenumbers={#1}, subparagraph/pagenumbers={#1}, %
1651     pagenumber/.alias=pagenumbers,%
1652   }%
1653 %% Specific to lof and lot -----
1654 \pgfqkeys{/itf/tocloft/lof}{%
1655   title/.store as code in cs=listfigurename,%
1656   listfigurename/.alias=title,%
1657   ref/.initial=\listfigurename,%
1658   ref/.default=\listfigurename,%
1659 }%
1660 \pgfqkeys{/itf/tocloft/lot}{%
1661   title/.store as code in cs=listtablename,%
1662   listtablename/.alias=title,%
1663   ref/.initial=\listtablename,%

```

```

1664      ref/.default=\listtablename,%  

1665 }%  

1666 \if@cftsubfigopt  

1667 \pgfqkeys{/itf/tocloft}{%  

1668   lof/pagenumbers/.style={figure/pagenumbers={#1},subfigure/pagenumbers={#1}},%  

1669     lof/pagenumber/.alias=lof/pagenumbers,%  

1670   lof/depth/.store in counter cs=c@lofdepth,%  

1671     lof/depth/.append code=\global\c@lofdepth\c@lofdepth\addtocontents{lof}{\c@lofdepth\the  

1672   lot/pagenumbers/.style={figure/pagenumbers={#1},subfigure/pagenumbers={#1}},%  

1673     lot/pagenumber/.alias=lot/pagenumbers,%  

1674   lot/depth/.store in counter cs=c@lotdepth,%  

1675     lot/depth/.append code=\global\c@lotdepth\c@lotdepth\addtocontents{lot}{\c@lotdepth\the  

1676 }%  

1677 \else  

1678 \pgfqkeys{/itf/tocloft}{%  

1679   lof/pagenumbers/.style={figure/pagenumbers={#1}},%  

1680     lof/pagenumber/.alias=lof/pagenumbers,%  

1681   lot/pagenumbers/.style={figure/pagenumbers={#1}},%  

1682     lot/pagenumber/.alias=lot/pagenumbers,%  

1683 }%  

1684 \fi

```

\itf@cft@sectionname This macro converts subsection into subsec etc.. because of tocloft command names: \cftbeforesubsecskip, \cftsubsecindent etc..

And the keys are either `subsec` or `subsection`.

```

1685 \def\itf@cft@name#1.#2.#3@nil{\cft#1\itf@cft@secname{#2}#3}  

1686 \def\itf@cft@secname#1{  

1687   \expandafter\itf@cft@sec@name\csname#1\endcsname{#1}%  

1688 }% \itf@cft@secname  

1689 \def\itf@cft@sec@name#1#2{  

1690   \ifx#1\chapter chap\else  

1691     \ifx#1\section sec\else  

1692       \ifx#1\subsection subsec\else  

1693         \ifx#1\subsubsection subsubsec\else  

1694           \ifx#1\paragraph para\else  

1695             \ifx#1\ subparagraph subpara\else  

1696               \ifx#1\ figure fig\else  

1697                 \ifx#1\subsubfigure subfig\else  

1698                   \ifx#1\table tab\else  

1699                     \ifx#1\subtable subtab\else  

1700                       #2\fi  

1701           \fi\fi\fi\fi\fi\fi\fi  

1702 }% \itf@cft@sec@name

```

### 21.11.1 Specific handlers for tocloft

```

1703 %% tocloft setup for tables sections :  

1704 \pgfkeys{  

1705   /handlers/.cft@def/.style={\pgfkeyscurrentpath/.store in cs=\itf@cft@name #1@nil},%  

1706   /handlers/.cft@code/.style={\pgfkeyscurrentpath/.store as code in cs=\itf@cft@name #1@nil},%  

1707   /handlers/.cft@length/.style={\pgfkeyscurrentpath/.store in skip cs=\itf@cft@name #1@nil},%  

1708 }

```

\itf@cft@newsection This macro create the keys for each sectionning command.

```

1709 \def\itf@cft@newsection#1#2{#1=part / chap / sec / subsec etc., #2 = #1 = toc | lof | lot  

1710   \pgfkeys{/itf/tocloft/#2/#1}{%  

1711     skip/.cft@length=before.#1.skip,%  

1712       beforeskip/.alias*=skip,%  

1713       top/.alias*=skip,%  

1714       indent/.cft@length=.#1.indent,%  

1715       font/.cft@code=.#1.font,%

```

```

1716     font/.append code={\pgfkeysalso{/itf/tocloft/#2/#1/pagenumber/font={##1}}},%
1717     format/.alias*=font,%
1718     number/width/.cft@length=.#1.numwidth,%
1719     numwidth/.alias*=number/width,%
1720     presnum/.cft@code=.#1.presnum,%
1721     number/before/.alias*=presnum,%
1722     aftersnum/.cft@code=.#1.aftersnum,%
1723     number/after/.alias*=aftersnum,%
1724     aftersnumb/.cft@code=.#1.aftersnumb,%
1725     title/before/.alias*=aftersnumb,%
1726     leader/.cft@def=.#1.leader,%
1727     leader/.default=\cftdotfill{\csname\itf@cft@name.#1.dotsep@\nil\endcsname},%
1728     leaders/.alias=leader,%
1729     dotsep/.code={\ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
1730       \csdef{\itf@cft@name.#1.dotsep@\nil}{\cftdotsep}%
1731     \else
1732       \itf@units@mu\itf@musk{##1}%
1733       \csedef{\itf@cft@name.#1.dotsep@\nil}{\itf@strip@mu\itf@musk}%
1734     \fi},%
1735     dotsep/.blank=\cftnodots,%
1736     nodots/.code=\csdef{\itf@cft@name.#1.dotsep@\nil}{\cftnodots},%
1737     nodots/.value forbidden,%
1738     pagefont/.cft@code=.#1.pagefont,%
1739     page/font/.alias*=pagefont,%
1740     pagenumber/font/.alias*=pagefont,%
1741     afterpnum/.cft@code=.#1.afterpnum,%
1742     page/after/.alias*=afterpnum,%
1743     pagenumber/after/.alias*=afterpnum,%
1744     pagenumbers/.on=@cftpnumon{\itf@cft@secname{#1}},%
1745     pagenumbers/.off=@cftpnumoff{\itf@cft@secname{#1}},%
1746     pagenumber/.alias=pagenumbers,%
1747   }% \pgfqkeys
1748 }% \itf@cft@newsection

1749 \@for\itf@tp:=part, chap, chapter, sec, section, subsec, subsection, %
1750   subsubsec, subsubsection, para, paragraph, subpara, subparagraph%
1751   \do{\expandafter\itf@cft@newsection\expandafter{\itf@tp}{toc}}%
1752 \if@cftsubfigopt % \@ifpackagewith{tocloft}{subfig} ??
1753   \@for\itf@tp:=fig, figure, subfig, subfigure%
1754   \do{\expandafter\itf@cft@newsection\expandafter{\itf@tp}{lof}}%
1755   \@for\itf@tp:=tab, table, subtab, subtable%
1756   \do{\expandafter\itf@cft@newsection\expandafter{\itf@tp}{lot}}%
1757 \else
1758   \@for\itf@tp:=fig, figure%
1759   \do{\expandafter\itf@cft@newsection\expandafter{\itf@tp}{lof}}%
1760   \@for\itf@tp:=tab, table%
1761   \do{\expandafter\itf@cft@newsection\expandafter{\itf@tp}{lot}}%
1762 \fi

@cftmaketocitle (patch)
@cftmakeloftitle (patch)
@cftmakelottitle (patch)
1763 \itf@patch\pretocmd@cftmaketocitle
1764   {\itf@tocloft\contentsname{toc}{tableofcontents}}
newlistentry (patch)
1765   {\PackageInfo{interfaces-tocloft}
1766     {Command \string@\cftmaketocitle\space successfully augmented}}
1767   {\PackageWarning{interfaces-tocloft}
1768     {Command \string@\cftmaketocitle\space could not be patched
1769      \MessageBreak label and bookmark keys for \string@\cftsetup\space won't work}}%
1770 \itf@patch\pretocmd@cftmakeloftitle
1771   {\itf@tocloft\listfigurename{lof}{listoffigures}}
1772   {\PackageInfo{interfaces-tocloft}
1773     {Command \string@\cftmakeloftitle\space successfully augmented}}
1774   {\PackageWarning{interfaces-tocloft}
1775     {Command \string@\cftmakeloftitle\space could not be patched}}

```

```

1776      \MessageBreak label and bookmark keys for \string\lofsetup\space won't work} }%
1777 \itf@patch\pretocmd@\cftmakelottitle
1778   {\itf@tocloft\listtablename{lot}{listoftables}}
1779   {\PackageInfo{interfaces-tocloft}
1780     {Command \string@\cftmakelottitle\space successfully augmented} }
1781   {\PackageWarning{interfaces-tocloft}
1782     {Command \string@\cftmakelottitle\space could not be patched
1783      \MessageBreak label and bookmark keys for \string\lotsetup\space won't work} }%
1784 \itf@patch\apptocmd\newlistentry
1785   {\itf@cft@newlistof{\#2}}
1786   {\PackageInfo{interfaces-tocloft}
1787     {Command \string\newlistentry\space successfully augmented} }
1788   {\PackageWarning{interfaces-tocloft}
1789     {Command \string\newlistentry\space could not be patched
1790      \MessageBreak \string\listofsetup\space will not work} }%

```

\itf@cft@label      \itf@cft@label insert the anchor for the table of contents and makes the label (if the key `label` of `\tocsetup`, `\lofsetup` etc. has been set to a non empty value.

```

1791 \def\itf@tocloft#1#2#3{%
1792   \expandafter\def\expandafter#1\expandafter{%
1793     \expandafter\itf@cft@label
1794       \expandafter#1\expandafter{#1}{#2}{#3}} }%
1795 }% \itf@tocloft
1796 \long\def\itf@cft@label#1#2#3#4{%
1797   \def#1{#2}% restore original
1798   \begingroup
1799   \pgfkeysifdefined{/itf/tocloft/#3/bookmark}
1800   {
1801     \raisedhyperdef{interfaces}{#4}{#2}%
1802     \pgfkeysgetvalue{/itf/tocloft/#3/bookmark/text}\itf@tp
1803     \ifx\itf@tp\empty\else
1804       \pgfkeysgetvalue{/itf/tocloft/#3/bookmark}@tempa
1805       \expandafter\bookmark\expandafter[%
1806         \expandafter{@tempa,dest=interfaces.#4}]%
1807         \pgfkeysvalueof{/itf/tocloft/#3/bookmark/text}}%
1808       \global\BKM@startatroottrue
1809   \fi}
1810   {#2}%
1811   \pgfkeysgetvalue{/itf/tocloft/#3/label}\itf@tp
1812   \ifx\itf@tp\empty\else
1813     \def@\currentHref{interfaces.#4}%
1814     \pgfkeysgetvalue{/itf/tocloft/#3/ref}@currentlabel
1815     \def@\currentlabelname{#2}%
1816     \label{\pgfkeysvalueof{/itf/tocloft/#3/label}}%
1817   \fi
1818   \endgroup
1819   \gdef\itf@tp{\def#1{#2}\let\itf@tp\undefined}% restore original
1820   \aftergroup\itf@tp % simple group introduced by tocloft to print \contentsname
1821 }% \itf@cft@label

1822 </tocloft>

```

## 21.12 † Interface `makecell` : `\repeatcell`

```

1823 /*makecell*/
1824 \ProvidesPackage{interfaces-makecell}
1825   [2010/11/16 v2.2 repeatcell to fill in tabulars (FC)]
1826 \RequirePackage{interfaces-base}
1827 \itf@Check@Package{makecell}
1828 \edef\itf@AtEnd{\catcode126 \the\catcode126}%
1829 \AtEndOfPackage{\itf@AtEnd}
1830 \catcode126 \active % 126 = ~
1831 \itf@pgfkeys

```

### 21.12.1 `countvalues`

```
\countvalues
```

```

1832 \def\countvalues#1{\number\numexpr
1833   \expandafter\itf@countvalues\expandafter#1,\countvalues@nnil
1834 }% \countvalues
1835 \def\itf@countvalues#1#2,#3@nnil{%
1836   \ifx\countvalues#3%
1837     +#1\expandafter\remove@to@nnil
1838   \else
1839     +1\expandafter\itf@countvalues
1840   \fi#3@nnil
1841 }% \itf@countvalues

```

```
\itf@getvalue
```

```

1842 \def\itf@getvalue#1#2{%
1843   \romannumerals-`q\expandafter\itf@getv@lue\expandafter#1,\itf@getvalue@nil{#2-1}@nnil
1844 }% \itf@getvalue
1845 \def\itf@getv@lue#1#2,#3@nil#4@nnil{%
1846   \ifnum\numexpr#1=\numexpr#4\relax
1847     #2\expandafter\remove@to@nnil
1848   \else\ifx\itf@getvalue#3%
1849     \expandafter\expandafter\expandafter\remove@to@nnil
1850   \else
1851     \expandafter\expandafter\expandafter\itf@getv@lue
1852   \fi\fi{#1+1}#3@nil#4@nnil
1853 }% \itf@getv@lue

```

### 21.12.2 `\repeatcell`

```
\repeatcell
```

```

1854 \newif\ifitf@celltranspose
1855 \def\itf@ampactive{%
1856   \def\tab{&}{%
1857     \lccode`~=\lowercase{\def~{\&}}{%
1858   }% \itf@ampactive
1859 \newrobustcmd*\repeatcell{\begingroup
1860   \let\tab\relax
1861   \catcode38\active
1862   \itf@repeatcell
1863 }% \repeatcell
1864 \newrobustcmd*\itf@repeatcell[3][]{%
1865   \ifblank{#1}{% alignment option ?
1866     {\let\itf@makecell@\firstofone{%
1867       {\def\itf@makecell##1{\noexpand\makecell[{#1}]{{##1}}}}{%
1868         \countdef\itf@col=100% local
1869         \countdef\itf@row=101% local
1870         \countdef\itf@cnt=102% local

```



```

1936      \fi
1937      \toks@\expandafter\expandafter\expandafter{%
1938          \expandafter\the\expandafter\toks@
1939          \itf@cell@init\itf@ampactive}%
1940      \itf@cell@insert}%
1941  }%
1942 \fi
1943 \itf@message{Code introduced by repeatcell in the \@currenvir:\the\toks@}%
1944 \expandafter\endgroup\the\toks@
1945 \ignorespaces
1946 }% \itf@repeatcell

\itf@message
1947 \let\itf@message\gobble

```

Now the keys for \repeatcell:

```

1948 \def\itf@cell@rowcolkey{%
1949     \removefromstart\pgfkeyscurrentname{row}%
1950     \ifin@\else
1951         \removefromstart\pgfkeyscurrentname{col}%
1952     \fi
1953     \ifin@
1954         \itf@number\itf@cnt\pgfkeyscurrentname\relax@nnil
1955         {\pgfkeyslet\pgfkeyscurrentkey\pgfkeyscurrentvalue}%
1956         \in@false
1957     \fi
1958     \ifin@\else
1959         \pgfkeysvalueof{/errors/unknown key/.@cmd}\pgfkeyscurrentkey
1960                                         \pgfkeyscurrentvalue
1961                                         \pgfeov
1962     \fi
1963 }% \itf@cell@rowcolkey
1964 \pgfqkeys{/itf/makecell/repeatcell}{%
1965     font/.store in=\itf@cellfont,%
1966     font/.unknown/.code=\itf@cell@rowcolkey,%
1967     text/.store in=\itf@celltext,%
1968     text/.default=\ifdefined\@C\@column\fi,%
1969     text/.unknown/.code=\itf@cell@rowcolkey,%
1970     cr/.initial=\\,%
1971     cr/.default=\crcr,cr/.blank=\\,end/.alias=cr,%
1972     cr/.unknown/.code=\itf@cell@rowcolkey,%
1973     rows/.code=\itf@number\itf@row#1\relax@nnil
1974         {\ifnum\itf@row<\z@\itf@row\@ne\fi}
1975         {\itf@row\z@
1976             \def\itf@rows{\#1}\itf@expandonceifcs\itf@rows},%
1977             rows/.default=1,rows/.blank=1,%
1978             @cols/.is private=interfaces-makecell,%
1979             column/.is private=interfaces-makecell,%
1980             row/.is private=interfaces-makecell,%
1981             transpose/.is if=\itf@celltranspose,%
1982             .unknown/.code=%
1983                 \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text \itf@cell@formulaerror{}\z@ \fi
1984                 \ifx\pgfkeyscurrentvalue\empty \itf@cell@formulaerror{}\@ne \fi
1985                 \pgfkeysifdefined{\pgfkeyscurrentkey/@calc}{%
1986                     {\itf@cell@formulaerror{}\tw@}
1987                     {\pgfkeyslet{\pgfkeyscurrentkey/@calc}\relax}%
1988                     \itf@cell@getformula{\#1}%
1989                     \pgfkeysEaddvalue{/itf/makecell/repeatcell/@calc}{%
1990                         {\the\toks@\expandafter\noexpand\csname\pgfkeyscurrentname\endcsname}%
1991                         \expandafter\let\csname\pgfkeyscurrentname\endcsname@\undefined,% (order of evaluation
1992 @reset/.code=% #1 are the columns
1993                         \itf@celltransposefalse

```

```

1994     \let\itf@celltext\@empty
1995     \let\itf@cellfont\@empty
1996     \pgfkeyssetvalue{/itf/makecell/repeatcell/cr}{\\}%
1997     \itf@row@ne
1998     \pgfkeyssetvalue{/itf/makecell/repeatcell/@calc}{%
1999         \itf@cell@numcalc\itf@col@column
2000         \itf@cell@numcalc\itf@row@row}%
2001     \itf@number\itf@col#1\relax\@nnil
2002         {\ifnum\itf@col<\z@\itf@col@ne\fi}
2003         {\itf@col\z@
2004             \def\itf@cols{\#1}\itf@expandonceifcs\itf@cols},%
2005 }% keys for \repeatcell

```

\itf@cell@getformula Determines whether the formula should be evaluated with \numexpr, with \\$\pgfmath\$ or \$\$\FPeval\$\$:

```

2006 \def\itf@cell@getformula#1{%
2007     \itf@cell@getf@rmula #1$$\itf@cell@getformula$\relax\@nnil
2008     \expandafter\ifx\expandafter\toks@\the\toks@
2009         \itf@cell@formulaerror{\#1}\thr@@
2010     \fi
2011 }% \itf@cell@getformula
2012 \def\itf@cell@getf@rmula #1##2##3##4##5##6{%
2013     \toks@{\toks@{%
2014         \ifx\itf@cell@getformula #4%
2015             \toks@{\itf@cell@numcalc{\#1}}%
2016         \else\ifx $#4%
2017             \ifx\itf@cell@getformula #6%
2018                 \ifdef\pgfmathparse\else\itf@cell@formulaerror{\#2}4\fi
2019                 \toks@{\itf@cell@pgfcalc{\#2}}%
2020             \else\ifx $#6%
2021                 \ifdef\FPeval\else\itf@cell@formulaerror{\#3}5\fi
2022                 \toks@{\itf@cell@FPcalc{\#3}}%
2023             \fi\fi
2024         \fi\fi\remove@to@nnil
2025 }% \itf@cell@getf@rmula
2026 \def\itf@cell@formulaerror#1#2{%
2027     \def\itf@tp{\#1}@onelvel@sanitize\itf@tp
2028     \ifcase#2% 0
2029         \PackageError{interfaces-makecell}{%
2030             {Missing formula for key \pgfkeyscurrentname!}@\ehd
2031         \or %1
2032         \PackageError{interfaces-makecell}{%
2033             {Computed key \pgfkeyscurrentname\space wes left blank!}@\ehd
2034         \or %2
2035         \PackageError{interfaces-makecell}{%
2036             {Duplicate definition for computed key \pgfkeyscurrentname
2037             \MessageBreak\string\repeatcell\space will redefine the key}@\ehd
2038         \or %3
2039         \PackageError{interfaces-makecell}{%
2040             {Incorrect formula in \string\repeatcell
2041             \MessageBreak Check: \itf@tp
2042             \MessageBreak Take care of commas and equal signs...}@\ehd
2043         \or %4
2044         \PackageError{interfaces-makecell}{%
2045             {Formula: \itf@tp
2046             \MessageBreak requires \string\pgfmathparse:\space\string\usetikz{pgfmath}}@\ehd
2047         \or %5
2048         \PackageError{interfaces-makecell}{%
2049             {Formula: \itf@tp
2050             \MessageBreak requires \string\FPeval:\space\string\usepackage{fp}}@\ehd
2051         \else
2052         \PackageError{interfaces-makecell}{%
2053             {Impossible value!}@\ehd
2054         \fi
2055 }% \itf@cell@formulaerror

```



```
\itf@cell@calc
\itf@cell@numcalc
\itf@cell@pgfcalc
\itf@cell@FPcalc
2056 \protected\def\itf@cell@calc#1#2{%
2057   \edef#1{\def\noexpand#1{#2}}%
2058   \toks@\expandafter\expandafter\expandafter{%
2059     \expandafter\the\expandafter\toks@#1}%
2060   #1%
2061 }% \itf@cell@calc
2062 \protected\def\itf@cell@numcalc#1#2{%
2063   \itf@cell@calc#2{\number\numexpr#1}%
2064 }% \itf@cell@numcalc
2065 \protected\def\itf@cell@pgfcalc#1#2{%
2066   \pgfmathparse{#1}%
2067   \itf@cell@calc#2\pgfmathresult
2068 }% \itf@cell@pgfcalc
2069 \protected\def\itf@cell@FPcalc#1#2{%
2070   \begingroup
2071   \setbox\z@\vbox{\%FPeval introduces some weird code, hence the calculation in a \vbox
2072     \FPeval\FPresult{#1}%
2073     \global\let\FPresult\FPresult}%
2074   \endgroup
2075   \itf@cell@calc#2\FPresult
2076 }% \itf@cell@FPcalc
```

`\itf@cell@insert` Insert a cell (into the temporary `\toks@` token register):

```
2077 \def\itf@cell@getifdefined#1#2{%
2078   \pgfkeysifdefined{/itf/makecell/repeatcell/#1}%
2079   {\pgfkeysgetvalue{/itf/makecell/repeatcell/#1}#2}%
2080   {\let#2\empty}%
2081 }%
2082 \def\itf@cell@insert{%
2083   \pgfkeysvalueof{/itf/makecell/repeatcell/@calc}%
2084   \ifx\R\empty\else
2085     \toks@\expandafter\expandafter\expandafter{%
2086       \expandafter\the\expandafter\toks@
2087       \expandafter\def\expandafter\r\expandafter{\r}}%
2088   \fi
2089   \ifx\C\empty\else
2090     \toks@\expandafter\expandafter\expandafter{%
2091       \expandafter\the\expandafter\toks@
2092       \expandafter\def\expandafter\c\expandafter{\c}}%
2093   \fi
2094   \itf@cell@getifdefined{font/row\the\itf@row}\itf@cellfontrow
2095   \itf@cell@getifdefined{font/col\the\itf@col}\itf@cellfontcol
2096   \itf@cell@getifdefined{text/row\the\itf@row}\itf@cellrow
2097   \itf@cell@getifdefined{text/col\the\itf@col}\itf@cellcol
2098   \toks@\expandafter\expandafter\expandafter{%
2099     \expandafter\the\expandafter\toks@\itf@cellfont}%
2100   \ifx\itf@cellrow\empty
2101     \ifx\itf@cellcol\empty
2102       \itf@cell\expandafter\expandafter\expandafter{%
2103         \expandafter\itf@cellfontcol\itf@cellfontrow}%
2104       \let\itf@tp\itf@celltext
2105     \else
2106       \itf@cell\expandafter{\itf@cellfontcol}%
2107       \let\itf@tp\itf@cellcol
2108     \fi
2109   \else
2110     \ifx\itf@cellcol\empty
2111       \itf@cell\expandafter{\itf@cellfontrow}%
2112       \let\itf@tp\itf@cellrow
2113     \else
2114       \itf@cell{}%
```

```

2115     \ifitf@celltranspose
2116     \edef\itf@tp{\noexpand\itf@diaghead(-2,1)
2117         {\expandonce\itf@cellfontrow\expandonce\itf@cellrow}
2118         {\expandonce\itf@cellfontcol\expandonce\itf@cellcol}}%
2119     \else
2120     \edef\itf@tp{\noexpand\itf@diaghead(-2,1)
2121         {\expandonce\itf@cellfontcol\expandonce\itf@cellcol}
2122         {\expandonce\itf@cellfontrow\expandonce\itf@cellrow}}%
2123     \fi
2124   \fi
2125 \fi
2126 \itf@cell\expandafter\expandafter\expandafter{%
2127     \expandafter\the\expandafter\itf@cell\itf@tp}%
2128 \edef\itf@tp{\itf@makecell{\the\itf@cell}}%
2129 \toks@\expandafter\expandafter\expandafter{%
2130     \expandafter\the\expandafter\toks@\itf@tp}%
2131 }% \itf@cell@insert

```

`\itf@diaghead` A helper macro to `makecell`'s `\diaghead`

```

2132 \def\itf@diaghead(#1)#2#3{%
2133   \begingroup
2134   \edef\itf@tp{%
2135     \noexpand\diaghead({#1}){\unexpanded{#2 #3}}%
2136     {\unexpanded{#2}}%
2137     {\unexpanded{#3}}}}%
2138   \expandafter\endgroup\itf@tp
2139 }% \itf@diaghead

```

### 21.12.3 colortbl: patch for `\CT@cellc`

`\CT@cellc`

```

2140 \AfterPackage*{colortbl}{%
2141 \def\CT@cellc[#2]#3{%
2142   #1 is either \@firstofone or \@gobble, [#2]#3 = color spec
2143   \xdef\CT@cell@color{\noexpand\CT@color#1{{#2}}}{#3}%
2144   \global\let\noexpand\CT@cell@color\relax}%
2144 }%

```

2145 </makecell>

## 21.13 Interface `enumitem` : `\setlist`

```

2146 {*enumitem}
2147 \ProvidesPackage{interfaces-enumitem}
2148   [2010/11/15 v2.1 more control on lists (FC)]
2149 \RequirePackage{interfaces-base}
2150 \itf@Check@Package{enumitem}
2151 \itf@pgfkeys

```

`\interitem` Like `amstext \intertext`, `\interitem` allows to add some non indented text inside a list environment:

```

2152 \providerobustcmd*\interitem{%
2153   \item[]\hskip-\leftmargin
2154 }% \interitem

```

```

\setlist (extension, star form)

2155 \renewrobustcmd*\setlist{\begingroup \@ifstar
2156   {\let\do\csgappto \itf@setlist}
2157   {\let\do\csgdef \itf@setlist}%

```

```

2158 }% \setlist
2159 \newrobustcmd*\itf@setlist[2][0]{%
2160   \let\enit@a@\empty
2161   \let\enit@b@\empty
2162   \@for\x:=#1\do{%
2163     \ifx\x@\empty\else
2164       \itf@number\count@\x\relax\@nnil
2165       {\edef\enit@b{\enit@b,\the\count@}}
2166       {\edef\enit@a{\enit@a,\x}}%
2167     \fi}%
2168   \ifx\enit@a@\empty\def\enit@a{list}\fi
2169   \ifx\enit@b@\empty\def\enit@b{0}\fi
2170   \@for\aa:=\enit@a\do{%
2171     \ifx\aa@\empty\else
2172       \@for\b@:=\enit@b\do{%
2173         \ifx\b@\empty\else
2174           \ifcsundef{end\aa}{%
2175             {\expandafter\itf@setlist@error\expandafter{\aa}}
2176             {\do{\enit@@\aa\romannumeral\b@}{,##2}}%
2177           \fi}%
2178         \fi}
2179   \endgroup
2180 }% \itf@setlist
2181 \def\itf@setlist@error#1{%
2182   \PackageError{interfaces-enumitem}{%
2183     {Environment ‘#1’ is undefined
2184     \MessageBreak Use \string\newlist\space before \string\setlist
2185     \MessageBreak Your command will be ignored}\@eha
2186 }% \itf@setlist@error

```

2187

2187 </enumitem>

## 21.14 † Interface truncate

```

2188 {*truncate}
2189 \ProvidesPackage{interfaces-truncate}

```

\trunc{text} \trunc{text} is like \truncate but leaves the text to its natural width if possible.

```

2190 \newrobustcmd\trunc{text}[3][\TruncateMarker]{% do nothing if width of #3 is less than #2
2191   \begin{group}
2192     \hbadness\maxdimen\hfuzz\vsiz
2193     \setbox\z@=\hbox to\dimexpr#2\relax{\ignorespaces#3\unskip}%
2194     \ifnum\badness>\@M
2195       \def\next{\truncate[{#1}]{#2}{#3}}%
2196     \else \def\next{#3}%
2197     \fi
2198     \expandafter\endgroup\next
2199 }% \trunc{text}

```

\htruncadjust

```

2200 \newrobustcmd\htruncadjust[1]{\@testopt
2201   {\itf@htruncadjust{#1}}{}%
2202 }% \htruncadjust
2203 \long\def\itf@htruncadjust#1[#2]#3{%
2204   #1=total width, #2=[x_1,...,x_n], #3=text_1,...,text_n
2205   \begin{group}
2206     \let\itf@trunc@width@\empty
2207     \let\itf@istruncated@\gobble % to remove the first comma
2208     \let\the@dimen\relax
2209     \count@\z@ \dimen@\z@
2210     \ifblank{#2}{% no optional [x_1,...,x_n] parameters
2211       %
2212       \@for\x:=#3\do{%

```

```

2212         \advance \count@ \@ne
2213         \setbox\z@=\hbox{\ignorespaces\x\unskip}%
2214         \dimen\count@ = \wd\z@
2215         \toks\count@\expandafter{\the\dimen\count@}%
2216         \advance\dimen@ \dimen\count@}%
2217     }%
2218   {%
2219     \@for\x:=#2\do{%
2220       \advance \count@ \@ne
2221       \toks\count@\expandafter{\x\p@}%
2222       \advance\dimen@ \x\p@
2223       \dimen\count@ -\@ne\p@}%
2224   }%
2225   \advance\count@\@ne \dimendef\itf@freespace=\count@
2226   \advance\count@\@ne \dimendef\itf@tempdim=\count@
2227   \advance\count@\@ne \dimendef\itf@sum=\count@
2228   \count@\z@      %%dimen0=\the\dimen@\quad\number\dimen@\par
2229   \itf@freespace\z@ % free space left when w_i < x_i/sum(x_i) * total
2230   \@for\x:=#3\do{%
2231     \advance \count@ \@ne
2232     \ifdim \dimen\count@<\z@ % not measured before
2233       \setbox\z@=\hbox{\ignorespaces\x\unskip}%
2234       \dimen\count@ = \wd\z@
2235     \fi
2236     \itf@tempdim = \itf@scaledimen{\#1}*\the\toks\count@\dimen@\relax
2237     \ifdim \dimen\count@ > \itf@tempdim
2238       \dimen\count@ = \itf@tempdim
2239       \edef\itf@istruncated{\itf@istruncated,\the\count@}%
2240       \advance\itf@sum by\the\toks\count@
2241     \else
2242       \advance\itf@freespace \itf@tempdim
2243       \advance\itf@freespace -\dimen\count@
2244     \fi
2245     \edef\itf@trunc@width{\itf@trunc@width\the\count@\{\the@dimen\the\count@\}}%
2246   }%
2247 %% Free space=\the\itf@freespace\par
2248 \ifdim\itf@freespace > \z@ % reinsert free space proportionally
2249   \ifx\itf@istruncated\@gobble\else
2250     \@for\x:=\itf@istruncated\do{%
2251       \advance\dimen\x by\itf@scaledimen\itf@freespace*\the\toks\x\itf@sum\relax
2252     }%
2253   \fi\fi
2254   \def\the@dimen{\the\dimen}%
2255   \edef\itf@trunc@width{\itf@trunc@width}%
2256   \expandafter\endgroup
2257   \expandafter\def
2258     \expandafter\itf@trunc@width
2259     \expandafter{\itf@trunc@width}%
2260 %% total=\the\dimexpr#1\relax
2261 }% \itf@htruncadjust
2262 \def\itf@scalefactor#1#2{\strip@pt\dimexpr#1*65536/#2\relax}
2263 \def\itf@scaledimen#1#2#3\relax{%
2264   \dimexpr\itf@scalefactor{#2}{#3}\dimexpr#1\relax\relax
2265 }% \itf@scaledimen

```

\htruncwidth

```

2266 \newcommand*\htruncwidth[1]{% purely expandable
2267   \expandafter\itf@htruncwidth\itf@trunc@width\relax\relax\@nnil{#1}%
2268 }% \htruncwidth
2269 \def\itf@htruncwidth#1#2#3\@nnil#4{%
2270   \ifx\relax#1\itf@trunc@nosuchwidth{#4}%
2271   \else\ifnum#1=#4
2272     #2%

```

```

2273 \else
2274   \itf@htruncwidth#3\@nnil{#4}%
2275 \fi\fi
2276 }% \itf@htruncwidth
2277 \def\itf@trunc@nosuchwidth#1{%
2278   \PackageError{interfaces-truncate}{%
2279     You requested the width for text nr#1
2280     \MessageBreak but there is no such text
2281     \MessageBreak Therefore no such width!}\@ehd
2282 }% \itf@trunc@nosuchwidth

2283 </truncate>

```

## 21.15 † interface fancyvrb

```

2284 <*fancyvrb>
2285 \ProvidesPackage{interfaces-fancyvrb}
2286   [2010/10/27 v1.2 Custom verbatim environments (FC)]
2287 \RequirePackage{interfaces-base}
2288 \itf@Check@Package{fancyvrb}
2289 \itf@pgfkeys
2290 \def\itf@tp#1{%
2291 \DefineVerbatimEnvironment{itf:VerbLines}{Verbatim}{%
2292   gobble=#1,frame=lines,framesep=6pt,fontseries=m}
2293 \DefineVerbatimEnvironment{itf:Verb}{Verbatim}{%
2294   gobble=#1,fontseries=m}
2295 \DefineVerbatimEnvironment{itf:Verb*}{Verbatim}{%
2296   gobble=#1,fontseries=m,commandchars=$()}

2297 }
2298 \ltx@ifpackageloaded{doc}{%
2299   \itf@tp{1}}
2300   \itf@tp{0}}
2301 \AtBeginDocument{%
2302 \ifdefined\VerbLines\else
2303 \ifdefined\endVerbLines\else
2304   \letcs\VerbLines{itf:VerbLines}%
2305   \letcs\endVerbLines{enditf:VerbLines}%
2306 \fi\fi
2307 \ifdefined\Verb\else\ifdefined\endVerb\else
2308   \letcs\Verb{itf:Verb}%
2309   \letcs\endVerb{enditf:Verb}%
2310 \fi\fi
2311 \ifcsname Verb*\endcsname\else
2312 \ifcsname endVerb*\endcsname\else
2313   \csletcs{Verb*}{itf:Verb*}%
2314   \csletcs{endVerb*}{enditf:Verb*}%
2315 \fi\fi
2316 }
2317 </fancyvrb>

```

## 21.16 † Interface ↗ appendix and the appendices environment

```

2318 <*appendix>
2319 \ProvidesPackage{interfaces-appendix}
2320   [2010/10/27 v1.2 easy control of appendices environment (FC)]
2321 \RequirePackage{interfaces-base}
2322 \itf@Check@Package{appendix}
2323 \itf@pgfkeys

\appendices
2324 \let\itf@appendices@pp\appendices
2325 \let\itf@endappendices@pp\endappendices
2326 \let\itf@appendices@pp\appendices
2327 \let\itf@endappendices@pp\endappendices

```

```

2328 \renewenvironment{appendices}[1][]{%
2329   \pgfqkeys{/itf/appendix/appendices}{#1}%
2330   \itf@appendices@pp
2331 }
2332 {\itf@endappendices@pp}%

2333 \pgfqkeys{/itf/appendix/appendices}{%
2334   title/.store as code in cs=appendixpagename,%
2335   title/.append code=\ifx\appendixpagename\empty \@dopage@ppfalse
2336           \else \@dopage@pptrue\fi,%
2337   title/.value required,%
2338   intoc/.is if=@dotoc@pp,%
2339   intoc/title/.store as code in cs=appendixtocname,%
2340   intoc/title/.append code=\ifx\appendixtocname\empty \@dotoc@ppfalse
2341           \else \@dotoc@pptrue\fi,%
2342   intoc/title/.default=\pgfkeyvalueof{/itf/appendix/appendices/title},%
2343   section/.is enum={section,chapter},%
2344   section/.append code=%
2345     \ifcase\pgfkeyvalueof{/itf/appendix/appendices/section/.number}\relax
2346       \@chapter@ppfalse\or \@chapter@pptrue\fi
2347       \if@chapter@pp\ifundefined{chapter}{%
2348         \PackageError{interfaces-appendix}
2349           {You asked chapter-style appendices but your
2350             \MessageBreak current class does not provide chapters!}\@eha
2351           \pgfkeyalso{/itf/appendix/appendices/section}}{}%
2352       \fi},%
2353   chapter/.style={section=chapter},%
2354   chapter/.value forbidden,%
2355 }% keys for appendices environment

2356 </appendix>

```

## 21.17 † Interface ↗ embedfile : \embedthisfile

```

2357 /*embedfile*/
2358 \ProvidesPackage{interfaces-embedfile}
2359   [2010/10/27 v1.2 easy macros to embed files into pdf (FC)]
2360 \RequirePackage{interfaces-base}
2361 \itf@Check@Package{embedfile}
2362 \newrobustcmd*\embedthisfile{\ifstar
2363   \embedthisfile@main
2364   \embedthisfile@all
2365 }% \embedthisfile
2366 \@onlypreamble\embedthisfile
2367 \def\embedthisfile@main{\embedfile{\jobname.tex}}
2368 \def\embedthisfile@all{%
2369   \embedthisfile@main
2370   \AfterPreamble{%
2371     \ltx@ifundefined{@listfiles}
2372       {\let\@listfiles\empty% to avoid: \global\let\@filelist\relax in \document
2373       }%
2374     }%
2375     \gpreto@enddocumenthook{%
2376       \let\do\itf@EmbedOne
2377       \expandafter\docslist\expandafter{\@filelist}%
2378     }%
2379 }% \embedthisfile@all
2380 \def\itf@EmbedOne#1{\itf@Embed@ne#1.tex\@nil}
2381 \def\itf@Embed@ne#1.tex#2\@nil{%
2382   \ltx@ifundefined{cp#1}
2383   {}
2384   {\ifblank{#2}{%
2385     {}%
2386     {\embedfile{#1.tex}}}}%

```

```
2387 }% \itf@Embed@ne
2388 </embedfile>
```

## 21.18 † Interface ↗ environ

```
2389 <*environ>
2390 \ProvidesPackage{interfaces-environ}
2391   [2010/07/23 v1.0 ReNewEnviron = RenewEnviron (FC)]
2392 \RequirePackage{interfaces-base}
2393 \AfterPackage{environ}{
2394   \ifundef\ReNewEnviron
2395     {\let\ReNewEnviron\RenewEnviron}
2396   {}%
2397 }%
2398 </environ>
```

## 21.19 † Interface umrand

```
2399 <*umrand>
2400 \ProvidesPackage{interfaces-umrand}
2401   [2010/07/23 v1.0 patch for umrand.sty ifnotempty (FC)]
2402 \RequirePackage{interfaces-base}
2403 \BeforePackage{umrand}{%
2404   \let\itf@newif\newif
2405   \def\itf@tp{\ifnotempty}%
2406   \def\newif#1{\def\x{#1}\ifx\x\itf@tp \expandafter\@gobble
2407     \else\expandafter\ltx@newif\fi{#1}}%
2408 }
2409 \AfterPackage{umrand}{%
2410   \let\newif\itf@newif
2411   \let\itf@newif@\undefined
2412 }
2413 </umrand>
```

## 21.20 † Interface ↗ scrlfile

```
2414 <*scrlfile>
2415 \ProvidesPackage{interfaces-scrlfile}
2416   [2010/10/27 v1.2 scrlfile hooks improved (FC)]
2417 \RequirePackage{interfaces-base}
```

\@pkgextension In L^AT_EX they are defined \@onlypreamble. So we use the ltxcmds equivalents:

\@clsextension

```
2418 \AfterEndPreamble{%
2419   \let\@pkgextension\ltx@pkgextension
2420   \let\@clsextension\ltx@clsextension
2421   \let@ifpackageloaded\ltx@ifpackageloaded
2422   \let@ifclassloaded\ltx@ifclassloaded
2423 }
```

\scr@load@hook

```
2424 \renewrobustcmd*\scr@load@hook[2]{%
2425   \csuse{#2-@#1}%
2426   \ifcsname#2-@#1-\string\keep\endcsname\else\global\csundef{#2-@#1}\fi
2427 }% \scr@load@hook
```

\scr@set@load@hook (optimisation)

```
2428 \renewrobustcmd*\scr@set@load@hook[2]{\csgappto{#2-@#1}}
```

\BeforeFile Addition of star forms *: the hooks are not canceled after input (in case a file is input more than once...)

\AfterFile

```

2429 \renewrobustcmd*\BeforeFile{@ifstar
2430   {\itf@s@LoadFile{before}}
2431   {\itf@LoadFile{before}}%
2432 }% \BeforeFile
2433 \renewrobustcmd*\AfterFile{@ifstar
2434   {\itf@s@LoadFile{after}}
2435   {\itf@LoadFile{after}}%
2436 }% \AfterFile

2437 \def\itf@s@LoadFile#1#2{%
2438   \global\cslet{\#2-@#1-\string\keep}{\empty}
2439   \scr@set@load@hook{#1}{#2}%
2440 }% \itf@s@LoadFile
2441 \def\itf@LoadFile#1#2{%
2442   \global\csundef{\#2-@#1-\string\keep}%
2443   \scr@set@load@hook{#1}{#2}%
2444 }% \itf@LoadFile

```

\@input (patch) This modification allows \@input to behaves like \InputIfFileExists as far as hooks are concerned.

```

2445 \renewcommand*\@input[1]{\IfFileExists{#1}
2446   {\scr@load@hook{before}{#1}%
2447     @@\input@filef@und
2448     \scr@load@hook{after}{#1}}%
2449   {\typeout{No file #1.}}%
2450 }% \@input

```

```
2451 </scrfile>
```

## 21.21 Interface pgfkeys ( pgfmanual)

### 21.21.1 interfaces-pgfkeys

```

2452 {*pgfkeys}
2453 \ProvidesPackage{interfaces-pgfkeys}
2454   [2010/11/16 v2.2 pgfkeys handlers definitions (FC)]
2455 \RequirePackage{interfaces-base}
2456 \ifdefined\pgfkeys@ifexecutehandler % pgf version 2.1
2457 \def\itf@pgfkeys{%
2458   \global\let\pgfkeys@unpack \itf@pgfkeys@unpack
2459   \global\let\pgfkeys@case@three \itf@pgfkeys@case@three
2460   \global\let\pgfkeys@case@three@handleall\itf@pgfkeys@case@three@handleall
2461 }% \itf@pgfkeys
2462 \else
2463 \def\itf@pgfkeys{%
2464   \global\let\pgfkeys@unpack \itf@pgfkeys@unpack
2465   \global\let\pgfkeys@case@three \itf@pgfkeys@case@three
2466   \global\let\pgfkeys@addvalue \itf@pgfkeys@addvalue
2467   \global\let\pgfkeys@exp@call \itf@pgfkeys@exp@call
2468 }% \itf@pgfkeys
2469 \fi
2470 \AtBeginDocument{%
2471   \itf@pgfkeys
2472   \let\itf@pgfkeys\relax
2473 }% AtBeginDocument

```

### Some new pgfkeys macros

\pgfkeysmeaning

```

2474 \providecommand*\pgfkeysmeaning[1]{%
2475   \pgfkeysifdefined{#1}

```

```

2476      {\expandafter\meaning\csname pgfk@#1\endcsname}
2477      {\meaning@\undefined}%
2478 }%

```

\pgfkeysvalueof@unexpanded

```

2479 \def\pgfkeysvalueof@unexpanded#1{%
2480   \unexpanded\expandafter\expandafter\expandafter{%
2481     \csname pgfk@#1\endcsname}%
2482 }% \pgfkeysvalueof@unexpanded

```

## Some pgfkeys macros redefinition/corrections

\pgfkeysaddvalue (bug in pgfkeysaddvalue for pgf v2.0, fixed in v2.1)

```

2483 \ifdefined\pgfkeys@ifexecutehandler \else % pgf v2.1: bug fixed
2484 \def\itf@pgfkeysaddvalue#1#2#3{%
2485   \edef\pgfkeys@global@temp{%
2486     \unexpanded{#2}%
2487     \ifcsname pgfk@#1\endcsname
2488       \unexpanded\expandafter\expandafter\expandafter{%
2489         \csname pgfk@#1\endcsname}%
2490     \fi
2491     \unexpanded{#3}}%
2492   \pgfkeyslet{#1}\pgfkeys@global@temp
2493 }% \itf@pgfkeysaddvalue
2494 \fi

```

\pgfkeysEsetvalue

\pgfkeysEaddvalue

```

2495 \ifdefined\pgfkeys@ifexecutehandler \long \fi % pgf version 2.1
2496 \def\pgfkeysEsetvalue#1#2{%
2497   \expandafter\edef\csname pgfk@#1\endcsname{#2}%
2498 }% \pgfkeysEsetvalue
2499 \ifdefined\pgfkeys@ifexecutehandler \long \fi % pgf v2.1
2500 \def\pgfkeysEaddvalue#1#2#3{%
2501   \csepreappto{pgfk@#1}{#2}{#3}
2502   \edef\pgfkeys@global@temp{%
2503     #2%
2504     \ifcsname pgfk@#1\endcsname
2505       \unexpanded\expandafter\expandafter\expandafter{%
2506         \csname pgfk@#1\endcsname}%
2507       \fi
2508     #3}%
2509   \pgfkeyslet{#1}\pgfkeys@global@temp
2509 }% \pgfkeysEaddvalue

```

\pgfkeys@exp@call Missed the braces around the argument (in case the argument contains a comma or an = sign...):

```

2510 %% missed a pair of braces around the argument in pgfkeys.code.tex ( code could break if it did )
2511 \ifdefined\pgfkeys@ifexecutehandler\else % pgf v2.1 (bug corrected in v2.1)
2512   \def\itf@pgfkeys@exp@call#1{\pgfkeysalso{\pgfkeyscurrentpath={#1}}}%
2513 \fi

```

## Handlers /blank

```

2514 %% The value is now checked by \itf@pgfkeys@default@blank@required:
2515 \ifdefined\pgfkeys@ifexecutehandler \long \fi % pgf v2.1
2516 \def\itf@pgfkeys@unpack#1=#2=#3\pgfkeys@stop{%
2517   \pgfkeys@spdef\pgfkeyscurrentkey{\#1}%
2518   \edef\pgfkeyscurrentkey{\pgfkeyscurrentkey}%
2519   \ifx\pgfkeyscurrentkey\pgfkeys@empty
2520     % Skip
2521   \else
2522     \pgfkeys@add@path@as@needed
2523     \pgfkeys@spdef\pgfkeyscurrentvalue{\#2}%
2524     \itf@pgfkeys@default@blank@required % <FC>
2525         \pgfkeyscurrentkey % <FC>
2526         \pgfkeys@case@one % <FC>
2527   \fi
2528 }% \itf@pgfkeys@unpack

```

\itf@pgfkeys@default@blank@required

Handle the case: `./@def` → `./@blank` → `./@def ...`

```

2529 \def\itf@pgfkeys@default@blank@required#1#2{%
2530   \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text% Hmm... no value
2531     \pgfkeysifdefined{\#1/.@def}%
2532       {\pgfkeys.getvalue{\#1/.@def}\pgfkeyscurrentvalue}%
2533       {}% no default, so leave it
2534   \fi
2535   \ifx\pgfkeyscurrentvalue\@empty
2536     \pgfkeysifdefined{\#1/.@blank}%
2537       {\pgfkeys.getvalue{\#1/.@blank}\pgfkeyscurrentvalue}%
2538       {}% no /.@blank specified
2539   \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text% in case /.@blank -> \pgfkeysnovalue
2540     \pgfkeysifdefined{\#1/.@def}%
2541       {\pgfkeys.getvalue{\#1/.@def}\pgfkeyscurrentvalue}%
2542       {}% no default, so leave it
2543   \fi\fi
2544   \ifx\pgfkeyscurrentvalue\pgfkeysvaluerequired
2545     \pgfkeysvalueof{/errors/value required/.@cmd}\pgfkeyscurrentkey
2546           \pgfkeyscurrentvalue
2547           \pgfeov
2548   \else
2549     #2%
2550   \fi
2551 }% \itf@pgfkeys@default@blank@required

```

## Allowing `/default` values (and `/blank` values) for handlers:

```

2552 %% To handle the cases of: /handler/.default= /handler/.blank= /handler/.value required,
2553 %% it is necessary to redefine \pgfkeys@case@three
2554 \def\itf@pgfkeys@case@three{%
2555   \pgfkeys@split@path
2556   \pgfkeysifdefined{/handlers/\pgfkeyscurrentname/.@cmd}%
2557   {%
2558     \itf@pgfkeys@default@blank@required{/handlers/\pgfkeyscurrentname}%
2559     {\pgfkeys.getvalue{/handlers/\pgfkeyscurrentname/.@cmd}\pgfkeys@code
2560       \expandafter\pgfkeys@code\pgfkeyscurrentvalue\pgfeov}%
2561   }
2562   \pgfkeys@unknown % no handler, thus the key is unknown
2563 }% \itf@pgfkeys@case@three
2564 \ifdefined\pgfkeys@ifexecutehandler
2565   \let\itf@pgfkeys@case@three@handleall \itf@pgfkeys@case@three
2566 \fi

```

## Loading interfaces modifications for pgfkeys

2567 \itf@pgfkeys

### Handlers definition: `./alias`, `./blank`

```
2568 \pgfkeys{%
2569   /handlers/.alias/.code=\itf@pgfkeys@ifaliasexists{\#1}{%
2570     \pgfkeysedef\pgfkeyscurrentpath{\noexpand\pgfkeysalso{\pgfkeysdefaultpath#1={##1}}},%
2571     /handlers/.alias/.value required,%%
2572   /handlers/.blank/.code=\pgfkeyssetvalue{\pgfkeyscurrentpath/.@blank}{\#1},%
2573     /handlers/.blank/.default=\pgfkeysnovalue,%
2574 }% ./alias and ./blank
2575 \def\itf@pgfkeys@ifaliasexists#1#2{%
2576   \pgfkeysifdefined{\pgfkeysdefaultpath#1/.@cmd}{\#2}{%
2577     \pgfkeysifdefined{\pgfkeysdefaultpath#1}{%
2578       {\#2}{%
2579         {\itf@pgfkeys@aliaserror{\#1}}}}%
2580 }% \itf@pgfkeys@ifaliasexists
2581 \def\itf@pgfkeys@aliaserror#1{%
2582   \PackageError{interfaces-pgfkeys}{%
2583     You want the key \pgfkeyscurrentpath\space%
2584     \MessageBreak to be an alias for \pgfkeysdefaultpath#1%
2585     \MessageBreak but the key \pgfkeysdefaultpath#1%
2586     \MessageBreak is not defined!}@\eha@\gobble
2587 }% \itf@pgfkeys@aliaserror
```

**Handlers definition: `./alias*`** For use with handlers `./store as glue` or `./store as glue in cs etc..` when a handler defines key with `*` version and/or `+` version and/or `-` versions.

Then, the `./alias*` handler is a facility to create aliases for starred `*`, `+`, `-` keys in a batch (as long as the alias exists, of course).

```
2588 \pgfkeys{%
2589   /handlers/.alias*/.code=\itf@pgfkeys@ifaliasexists{\#1}{%
2590     \pgfkeysedef\pgfkeyscurrentpath{\noexpand\pgfkeysalso{\pgfkeysdefaultpath#1={##1}}}}%
2591     @tfor\x:=-*\do{%
2592       \pgfkeysifdefined{\pgfkeysdefaultpath#1\x/.@cmd}{%
2593         \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2594           \noexpand\pgfkeysalso{\pgfkeysdefaultpath#1\x={##1}}}}%
2595     }%
2596   },%
2597   /handlers/.alias*/.value required,%
2598 }% ./alias*
```

### Handlers definition: `./is private`

```
2599 \pgfkeys{%
2600   /handlers/.is private/.code=\pgfkeysedef\pgfkeyscurrentpath{%
2601     \PackageError{\#1}{Key \pgfkeyscurrentkey\space is private%
2602     \MessageBreak and should not be modified}@\ehd},%
2603   /handlers/.is private to class/.code=\pgfkeysedef\pgfkeyscurrentpath{%
2604     \ClassError{\#1}{Key \pgfkeyscurrentkey\space is private%
2605     \MessageBreak and should not be modified}@\ehd}%
2606 }% ./is private
```

**Handlers definition: `/true` `/false` `/on` `/off`**

```

2607 \def\itf@pgfkeys@choice[#1]#2#3{%
2608   \pgfkeys{\pgfkeyscurrentpath/.cd,%
2609     .is choice,%
2610     #2/.code={#3},%
2611     .default/.expanded=\@firstoftwo#1}%
2612 }% \itf@pgfkeys@choice
2613 \pgfkeys{%
2614   /handlers/.true/.code=\itf@pgfkeys@choice[{true}{false}]{true}{#1},%
2615   /handlers/.false/.code=\itf@pgfkeys@choice[{true}{false}]{false}{#1},%
2616   /handlers/.on/.code=\itf@pgfkeys@choice[{on}{off}]{on}{#1},%
2617   /handlers/.off/.code=\itf@pgfkeys@choice[{on}{off}]{off}{#1},%
2618 }% /.true /.false

```

**Handlers definition: `/store in cs` `/estore in cs` etc.**

```

2619 \pgfkeys{%
2620   /handlers/.store in cs/.code=\pgfkeysdef\pgfkeyscurrentpath{\csdef{#1}{##1}},%
2621   /handlers/.estore in cs/.code=\pgfkeysdef\pgfkeyscurrentpath{\csedef{#1}{##1}},%
2622 }% /.store in cs

```

**Handler definition: `/store as code`**

```

2623% /.store as code handlers
2624 \pgfkeys{%
2625   /handlers/.store as code/.code={%
2626     \pgfkeysedef\pgfkeyscurrentpath{%
2627       \noexpand\pgfkeyslet{\pgfkeyscurrentpath}\noexpand\pgfkeyscurrentvalue}%
2628     \pgfkeyslet\pgfkeyscurrentpath@\empty
2629     \pgfkeyslet{\pgfkeyscurrentpath/.@def}\@empty
2630     @tfor\x:=#1\do{%
2631       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2632         \noexpand\pgfkeysalso{\pgfkeyscurrentpath/.expanded={%
2633           \noexpand\pgfkeysvalueof@unexpanded{\pgfkeyscurrentpath}%
2634           \noexpand\unexpanded{##1}}}}%
2635       \pgfkeyslet{\pgfkeyscurrentpath\x/.@def}\@empty
2636     }%,%
2637   /handlers/.store as code/.default=*,%
2638 }% /.store as code

```

**Handlers definition: `/estore as code`**

```

2639 \pgfkeys{%
2640   /handlers/.estore as code/.code={%
2641     \pgfkeysedef\pgfkeyscurrentpath{%
2642       \noexpand\pgfkeysEsetvalue{\pgfkeyscurrentpath}{##1}}%
2643     \pgfkeyslet\pgfkeyscurrentpath@\empty
2644     @tfor\x:=#1\do{%
2645       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2646         \noexpand\pgfkeysalso{\pgfkeyscurrentpath/.expanded={%
2647           \noexpand\pgfkeysvalueof@unexpanded{\pgfkeyscurrentpath}%
2648           \noexpand\unexpanded{##1}}}}%
2649     }%,%
2650   /handlers/.estore as code/.default=*,%
2651 }% /.estore as code

```

**Handlers definition: `./store as code in cs=<cs-name>`**

```

2652 \pgfkeys{%
2653   /handlers/.store as code in cs/.code={%
2654     \pgfkeysdef\pgfkeyscurrentpath{%
2655       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
2656       \else\csdef{\#1}{##1}\fi}%
2657     \@tfor\x:=*+\do{%
2658       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2659         \noexpand\pgfkeysalso{\pgfkeyscurrentpath/.expanded={%
2660           \noexpand\ifcsname#1\endcsname
2661             \noexpand\expandafter\noexpand\expandonce
2662               \noexpand\csname#1\endcsname
2663             \noexpand\fi
2664             \noexpand\unexpanded{##1}}}}%
2665       \pgfkeyslet{\pgfkeyscurrentpath\x/.@def}\@empty
2666     }},%
2667   /handlers/.store as code in cs/.value required,%
2668 }% /.store as code in cs
2669 \pgfkeys{%
2670   /handlers/.estore as code in cs/.code={%
2671     \pgfkeysdef\pgfkeyscurrentpath{%
2672       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
2673       \else\csedef{\#1}{##1}\fi}%
2674     \@tfor\x:=*+\do{%
2675       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2676         \noexpand\pgfkeysalso{\pgfkeyscurrentpath/.expanded={%
2677           \noexpand\ifcsname#1\endcsname
2678             \noexpand\expandafter\noexpand\expandonce
2679               \noexpand\csname#1\endcsname
2680             \noexpand\fi
2681             \noexpand\unexpanded{##1}}}}%
2682     },% default if \pgfkeysnovalue which expands to \@empty
2683   /handlers/.estore as code in cs/.value required,%
2684 }% /.estore as code in cs

```

**Handlers definition: `./store as color, ./store as color command`**

```

2685 \protected\def\itf@pgfk@color#1#2@nil{%
2686   \ifx@let@token\color
2687     \def\pgfkeyscurrentvalue{#2}%
2688   \else\ifx@let@token\normalcolor
2689     \def\pgfkeyscurrentvalue{default}%
2690   \else\ifx@let@token\pgfkeysnovalue
2691     \def\pgfkeyscurrentvalue{default}%
2692   \fi\fi\fi
2693 }% \itf@pgfk@color
2694 \pgfkeys{%
2695   /handlers/.store as color/.code=%
2696     \pgfkeysedef\pgfkeyscurrentpath{%
2697       \futurelet\noexpand\@let@token\itf@pgfk@color
2698         ##1\noexpand\pgfkeysnovalue\noexpand\@nil
2699       \noexpand\pgfkeyslet{\pgfkeyscurrentpath}\noexpand\pgfkeyscurrentvalue}%
2700 }% /.store as color

```

**Handlers definition: `./store as color in cs=<cs-name>`**

```

2701 \pgfkeys{%
2702   /handlers/.store as color in cs/.code=%
2703   \pgfkeysedef\pgfkeyscurrentpath{%
2704     \futurelet\noexpand\@let@token\itf@pgfk@color
2705     ##1\noexpand\pgfkeysnovalue\noexpand\@nil
2706     \cslet{\#1}\pgfkeyscurrentvalue}%
2707 }% /.store as color in cs

```

**Handlers definition: `./store as color command`**

```

2708 \protected\def\itf@pgfk@colorcmd#1@nil{%
2709   \ifcat @\noexpand\@let@token
2710     \def\pgfkeyscurrentvalue{\color{\#1}}%
2711   \else\ifcat 0\@expand\@let@token
2712     \def\pgfkeyscurrentvalue{\color{\#1}}%
2713   \else\ifx\@let@token\@nil
2714     \let\pgfkeyscurrentvalue\normalcolor
2715   \fi\fi\fi
2716 }% \itf@pgfk@colorcmd
2717 \pgfkeys{%
2718   /handlers/.store as color command/.code=%
2719   \pgfkeysedef\pgfkeyscurrentpath{%
2720     \futurelet\noexpand\@let@token\itf@pgfk@colorcmd
2721     ##1\noexpand\@nil
2722     \noexpand\pgfkeyslet{\pgfkeyscurrentpath}\noexpand\pgfkeyscurrentvalue}%
2723     \pgfkeyssetvalue{\pgfkeyscurrentpath/.@def}\normalcolor
2724 }% /.store as color command

```

**Handlers definition: `./store as color command in cs=<cs-name>`**

```

2725 \pgfkeys{%
2726   /handlers/.store as color command in cs/.code=%
2727   \pgfkeysedef\pgfkeyscurrentpath{%
2728     \futurelet\noexpand\@let@token\itf@pgfk@colorcmd
2729     ##1\noexpand\@nil
2730     \cslet{\#1}\noexpand\pgfkeyscurrentvalue}%
2731     \pgfkeyssetvalue{\pgfkeyscurrentpath/.@def}\normalcolor
2732 }% /.store as color command in cs

```

**Handlers definition `./store as number`**

```

2733 \pgfkeys{%
2734   /handlers/.store as number/.code={%
2735     \pgfkeysedef\pgfkeyscurrentpath{%
2736       \noexpand\pgfkeyssetvalue{\pgfkeyscurrentpath}{\numexpr##1\relax}%
2737     \pgfkeyslet\pgfkeyscurrentpath\z@
2738     \pgfkeyslet{\pgfkeyscurrentpath/.@def}\z@
2739     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@
2740     \atfor\x:=#1\do{%
2741       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2742         \noexpand\pgfkeysalso{\pgfkeyscurrentpath/.expanded=%
2743           \noexpand\pgfkeysvalueof{unexpanded{\pgfkeyscurrentpath}}%
2744           \x
2745           \numexpr\unexpanded{\#1}\relax}%
2746         \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x\@ne\else\z@\fi}%
2747         \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x\@ne\else\z@\fi}%
2748       },%
2749     /handlers/.store as number/.default=+-*,%
2750   }% /.store as number

```

**Handlers definition: `./estore as number`**

```

2751 \pgfkeys{%
2752   /handlers/.estore as number/.code={%
2753     \pgfkeysedef\pgfkeyscurrentpath{%
2754       \noexpand\pgfkeysetvalue{\pgfkeyscurrentpath}{%
2755         \noexpand\number\numexpr##1\relax}%
2756       \pgfkeyslet\pgfkeyscurrentpath\z@%
2757       \pgfkeyslet{\pgfkeyscurrentpath/.@def}\z@%
2758       \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@%
2759     @tfor\x:=#1\do{%
2760       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2761         \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
2762           \noexpand\pgfkeysvalueof{\pgfkeyscurrentpath}%
2763           \x
2764           \numexpr##1\relax}%
2765       }%,%
2766     /handlers/.estore as number/.default=+-*,%
2767   }% /.estore as number

```

**`./store as number in cs` Handlers definition: `./store as number in cs`**

```

2768 \pgfkeys{%
2769   /handlers/.store as number in cs/.code={%
2770     \pgfkeysedef\pgfkeyscurrentpath{%
2771       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
2772       \else\csdef{\#1}{\numexpr##1\relax}\fi}%
2773     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@%
2774     @tfor\x:=+-*\do{%
2775       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2776         \letcs\noexpand\pgfkeys@temp{%
2777           \noexpand\ifcsname#1\endcsname
2778           #1\noexpand\else\if*\x @ne\else z@\fi\noexpand\fi}%
2779         \edef\noexpand\pgfkeys@temp{%
2780           \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
2781             \noexpand\expandonce\noexpand\pgfkeys@temp
2782             \x
2783             \numexpr\noexpand\unexpanded{##1}\relax}%
2784           }\noexpand\pgfkeys@temp}%
2785         \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else z@\fi}%
2786         \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else z@\fi}%
2787     }%,%
2788   /handlers/.store as number in cs/.value required,%
2789 }% /.store as number in cs

```

**Handlers definition: `./estore as number in cs`**

```

2790 \pgfkeys{%
2791   /handlers/.estore as number in cs/.code={%
2792     \pgfkeysedef\pgfkeyscurrentpath{%
2793       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
2794       \else\csedef{\#1}{\number\numexpr##1\relax}\fi}%
2795     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@%
2796     @tfor\x:=+-*\do{%
2797       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2798         \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
2799           \noexpand\csname\noexpand\ifcsname#1\endcsname
2800           #1\noexpand\else\if*\x @ne\else z@\fi\noexpand\fi\endcsname
2801           \x
2802           \numexpr##1\relax}%
2803         \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else z@\fi}%
2804         \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else z@\fi}%
2805     }%,%
2806   /handlers/.estore as number in cs/.value required,%

```

```
2807 }% /.estore as number in cs
```

### Handlers definition: `./store in counter cs`

```
2808 \pgfkeys{%
2809   /handlers/.store in counter cs/.code={%
2810     \pgfkeysdef\pgfkeyscurrentpath{%
2811       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
2812       \else\csname#1\endcsname\numexpr##1\relax\fi}%
2813     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@
2814   @tfor\x:=+-*\do{%
2815     \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2816       \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
2817         \noexpand\csname\noexpand\ifcsname#1\endcsname
2818         #1\noexpand\else\if*\x @ne\else z@\fi\noexpand\fi\endcsname
2819         \x
2820         \numexpr##1\relax}%
2821       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else z@\fi}%
2822       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\@ne\else z@\fi}%
2823     }},%
2824   /handlers/.store in counter cs/.value required,%
```

2825 }% /.store in counter cs

### Handlers definition: `./store as glue`

```
2826 \pgfkeys{%
2827   /handlers/.store as glue/.code={%
2828     \pgfkeysedef\pgfkeyscurrentpath{%
2829       \noexpand\pgfkeyssetvalue{\pgfkeyscurrentpath}{\glueexpr##1\relax}}%
2830     \pgfkeyslet\pgfkeyscurrentpath\z@skip
2831     \pgfkeyslet{\pgfkeyscurrentpath/.@def}\z@skip
2832     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@skip
2833   @tfor\x:=#1\do{%
2834     \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2835       \noexpand\pgfkeysalso{\pgfkeyscurrentpath/.expanded=%
2836         \noexpand\pgfkeysvalueof{unexpanded{\pgfkeyscurrentpath}}%
2837         \x
2838         \if*\x\noexpand\itf@units@scale##1!
2839         \else\glueexpr\noexpand\unexpanded{##1}\relax\fi}%
2840       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else z@skip\fi}%
2841       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else z@skip\fi}%
2842     }},%
2843   /handlers/.store as glue/.default=+-*,%
2844 }% /.store as glue
```

### Handlers definition: `./estore as glue`

```
2845 \pgfkeys{%
2846   /handlers/.estore as glue/.code={%
2847     \pgfkeysedef{\pgfkeyscurrentpath}{%
2848       \noexpand\pgfkeysEsetvalue{\pgfkeyscurrentpath}{%
2849         \noexpand\the\glueexpr##1\relax}}%
2850     \pgfkeyslet\pgfkeyscurrentpath\z@skip
2851     \pgfkeyslet{\pgfkeyscurrentpath/.@def}\z@skip
2852     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@skip
2853   @tfor\x:=#1\do{%
2854     \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2855       \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
2856         \noexpand\pgfkeysvalueof{\pgfkeyscurrentpath}\x
2857         \if*\x\noexpand\itf@units@scale##1!
2858         \else\glueexpr##1\relax\fi}%
2859       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else z@skip\fi}%
2860     }},%
```

```

2860     \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else\z@skip\fi}%
2861     },%
2862     /handlers/.estore as glue/.default=+-*,%
2863 }% /.estore as glue

```

### Handlers definition: `./store as dimen`

```

2864 \pgfkeys{%
2865     /handlers/.store as dimen/.code={%
2866         \pgfkeysedef\pgfkeyscurrentpath{%
2867             \noexpand\pgfkeyssetvalue{\pgfkeyscurrentpath}{\dimexpr##1\relax}}%
2868             \pgfkeyslet\pgfkeyscurrentpath\z@%
2869             \pgfkeyslet{\pgfkeyscurrentpath/.@def}\z@%
2870             \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@%
2871             @tfor\x:=#1\do{%
2872                 \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2873                     \noexpand\pgfkeysalso{\pgfkeyscurrentpath/.expanded={%
2874                         \noexpand\pgfkeysvalueof@unexpanded{\pgfkeyscurrentpath}%
2875                         \x%
2876                         \if*\x \noexpand\itf@units@scale##1!
2877                         \else\dimexpr\noexpand\unexpanded{##1}\relax\fi}}}%
2878             \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else\z@\fi}%
2879             \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else\z@\fi}%
2880         }%,%
2881     /handlers/.store as dimen/.default=+-*,%
2882 }% /.store as dimen

```

### Handlers definition: `./estore as dimen`

```

2883 \pgfkeys{%
2884     /handlers/.estore as dimen/.code={%
2885         \pgfkeysedef\pgfkeyscurrentpath{%
2886             \noexpand\pgfkeysEsetvalue{\pgfkeyscurrentpath}{%
2887                 \noexpand\the\dimexpr##1\relax}}%
2888             \pgfkeyslet\pgfkeyscurrentpath\z@%
2889             \pgfkeyslet{\pgfkeyscurrentpath/.@def}\z@%
2890             \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@%
2891             @tfor\x:=#1\do{%
2892                 \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2893                     \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
2894                         \noexpand\pgfkeysvalueof{\pgfkeyscurrentpath}\x%
2895                         \if*\x \noexpand\itf@units@scale##1!
2896                         \else \dimexpr##1\relax\fi}}%
2897             \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else\z@\fi}%
2898             \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else\z@\fi}%
2899         }%,%
2900     /handlers/.estore as dimen/.default=+-*,%
2901 }% /.estore as dimen

```

### Handlers definition: `./store as glue in cs=<cs-name>`

```

2902 \pgfkeys{%
2903     /handlers/.store as glue in cs/.code={%
2904         \pgfkeysedef\pgfkeyscurrentpath{%
2905             \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text%
2906             \else\csdef{#1}{\glueexpr##1\relax}\fi}%
2907             \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@skip%
2908             @tfor\x:=+-*\do{%
2909                 \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2910                     \letcs\noexpand\pgfkeys@temp{%
2911                         \noexpand\ifcsname#1\endcsname%
2912                         #1\noexpand\else z@skip\noexpand\fi}}%

```

```

2913         \edef\noexpand\pgfkeys@temp{%
2914             \noexpand\pgfkeysalso{\pgfkeyscurrentpath={%
2915                 \noexpand\expandonce\noexpand\pgfkeys@temp
2916                 \x
2917                 \if*\x\noexpand\itf@units@scale##1!
2918                 \else\glueexpr\noexpand\unexpanded{##1}\relax\fi}}%
2919             }\noexpand\pgfkeys@temp}%
2920             \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else\z@skip\fi}%
2921             \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else\z@skip\fi}%
2922         }},%
2923     /handlers/.store as glue in cs/.value required,%
2924 }% /.store as glue in cs

```

### Handlers definition: `/.estore as glue in cs=<cs-name>`

```

2925 \pgfkeys{%
2926     /handlers/.estore as glue in cs/.code={%
2927         \pgfkeysdef\pgfkeyscurrentpath{%
2928             \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
2929             \else\csedef{#1}{\the\glueexpr##1\relax}\fi}%
2930         \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@skip
2931         \@tfor\x:=+-*\do{%
2932             \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2933                 \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
2934                     \noexpand\csname\noexpand\ifcsname#1\endcsname
2935                         #1\noexpand\else z@skip\noexpand\fi\endcsname
2936                     \x
2937                     \if*\x\noexpand\itf@units@scale##1!
2938                     \else\glueexpr##1\relax\fi}}%
2939             \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else\z@skip\fi}%
2940             \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else\z@skip\fi}%
2941         }},%
2942     /handlers/.etore as glue in cs/.value required,%
2943 }% /.estore as glue in cs

```

### Handlers definition: `/.store in skip cs`

This is a variant of `/.store as glue in cs` when the control sequence in a `\skip` (*i.e.* a length). The skip is given by its name, not as a control sequence.

```

2944 \pgfkeys{%
2945     /handlers/.store in skip cs/.code={%
2946         \pgfkeysdef\pgfkeyscurrentpath{%
2947             \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
2948             \else\csname#1\endcsname\glueexpr##1\relax\fi}%
2949         \pgfkeyslet{/pgfkeyscurrentpath/.@blank}\z@skip
2950         \@tfor\x:=+-*\do{%
2951             \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2952                 \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
2953                     \csname#1\endcsname % should have been defined as a skip (not expandable)
2954                     \x
2955                     \if*\x\noexpand\itf@units@scale##1!
2956                     \else\glueexpr##1\relax\fi}}%
2957             \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else\z@skip\fi}%
2958             \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else\z@skip\fi}%
2959         }},%
2960     /handlers/.store in skip cs/.value required,%
2961 }% /.store in skip cs

```

**Handlers definition: `/store as dimen in cs=<cs-name>`**

```

2962 \pgfkeys{%
2963   /handlers/.store as dimen in cs/.code={%
2964     \pgfkeysdef\pgfkeyscurrentpath{%
2965       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
2966       \else\csdef{\#1}{\dimexpr##1\relax}\fi}%
2967     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@
2968     @tfor\x:=+-*\do{%
2969       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2970         \letcs\noexpand\pgfkeys@temp{%
2971           \ifcsname#1\endcsname#1\else z@\fi}%
2972         \edef\noexpand\pgfkeys@temp{%
2973           \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
2974             \noexpand\expandonce\noexpand\pgfkeys@temp
2975             \x
2976             \if*\x\noexpand\itf@units@scale##1!
2977             \else\dimexpr\noexpand\unexpanded{##1}\fi}%
2978           }\noexpand\pgfkeys@temp}%
2979         \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else\z@\fi}%
2980         \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else\z@\fi}%
2981     }},%
2982   /handlers/.store as dimen in cs/.value required,%
2983 }% /.store as dimen in cs {csname}{+-}

```

**Handlers definition: `/estore as dimen in cs=<cs-name>`**

```

2984 \pgfkeys{%
2985   /handlers/.estore as dimen in cs/.code={%
2986     \pgfkeysdef\pgfkeyscurrentpath{%
2987       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
2988       \else \@defaultunits\itf@sk ##1pt\relax\@nnil
2989       \csedef{\#1}{\the\dimexpr\itf@sk\relax}\fi}%
2990     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@
2991     @tfor\x:=+-*\do{%
2992       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2993         \noexpand\ifcsname#1\endcsname
2994           \noexpand@defaultunits\itf@sk
2995           \noexpand\csname#1\endcsname pt\relax\noexpand\@nnil
2996           \noexpand\else\itf@sk\z@\noexpand\fi
2997           \if*\x
2998             \itf@sk\dimexpr\itf@sk*\noexpand\itf@units@scale##1!\relax
2999           \else
3000             \noexpand@defaultunits@tempdima ##1pt\relax\noexpand\@nnil
3001             \advance\itf@sk\x@tempdima
3002           \fi
3003           \noexpand\pgfkeysalso{\pgfkeyscurrentpath=\itf@sk}%
3004         \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else\z@\fi}%
3005         \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else\z@\fi}%
3006     }},%
3007   /handlers/.etore as dimen in cs/.value required,%
3008 }% /.estore as dimen in cs

```

**Handlers definition: `./store in dimen cs`** This is a variant of `./store as dimen in cs` when the control sequence in a `\dimen`. The `dimen` is given by its name, not as a control sequence.

```

3009 \pgfkeys{%
3010   /handlers/.store in dimen cs/.code={%
3011     \pgfkeysdef\pgfkeyscurrentpath{%
3012       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
3013       \else\csname#1\endcsname\dimexpr##1\relax\fi}%
3014     \pgfkeyslet{/pgfkeyscurrentpath/.@blank}\z@
3015     \@tfor\x:=+-*\do{%
3016       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
3017         \if*\x \else
3018           \noexpand@\defaultunits\itf@sk ##1pt\relax\noexpand@nnil
3019         \fi
3020           \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
3021             \csname#1\endcsname % should have been defined as a dimen (not expandable)
3022             \x
3023             \if*\x \noexpand\itf@units@scale##1!
3024             \else\itf@sk\fi}%
3025           \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@\ne\else\z@\fi}%
3026           \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@\ne\else\z@\fi}%
3027     }},%
3028   /handlers/.store in dimen cs/.value required,%
```

3029 }% /.store in dimen cs

**Handlers definition: `./is multiple choice`**

```

3030 \pgfkeys{%
3031   /handlers/.is multiple choice/.code=%
3032     \pgfkeys{\pgfkeyscurrentpath/.cd,%
3033       .unknown/.style={/errors/unknown choice value=\pgfkeyscurrentkey\pgfkeyscurrentvalue
3034       .code=\expandafter\pgfkeys\expandafter{\pgfkeyscurrentkey/.cd,####1}%
3035     },%
3036   /handlers/.is multiple choice/.value forbidden,%
```

3037 }% /.is multiple choice

**Handlers definition: `./is enum`**

```

3038 \pgfkeys{%
3039   /handlers/.is enum/.code={%
3040     \itf@cnt\z@
3041     \@for\x:=\pgfkeyscurrentvalue\do{%
3042       \pgfkeysedef{\pgfkeyscurrentpath/\x}{%
3043         \noexpand\pgfkeyssetvalue{\pgfkeyscurrentpath}{\x}%
3044         \noexpand\pgfkeyssetvalue{\pgfkeyscurrentpath/.number}{\the\itf@cnt}%
3045       \ifnum\itf@cnt=\z@
3046         \pgfkeyslet{\pgfkeyscurrentpath/.@def}\x
3047         \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\x
3048       \fi
3049       \advance\itf@cnt\@ne
3050     }%
3051     \pgfkeysalso{\pgfkeyscurrentpath/.is choice},%
3052   /handlers/.is enum/.value required,%
```

3053 }% /.is enum

3054

## 21.22 † Interface etoolbox - etoolbox-like commands

```

3055 {*etoolbox}
3056 \ProvidesPackage{interfaces-etoolbox}
3057   [2010/10/27 v1.2 some more etoolbox-like commands (FC)]
3058 \RequirePackage{interfaces-base}
3059 \begingroup\catcode`\&=3
3060   \long\gdef\etb@doitem#1&{\do{#1}}
3061 \endgroup

\preappto
\epreappto
\gpreappto 3062 \providerobustcmd\preappto[3]{%
3063   \edef#1{\unexpanded{\#2}\ifdefined#1\expandonce#1\fi\unexpanded{\#3}}%
3064 }% \preappto
3065 \providerobustcmd\epreappto[3]{%
3066   \edef#1{\#2\ifdefined#1\expandonce#1\fi\#3}%
3067 }% \epreappto
3068 \providerobustcmd*\gpreappto{\global\preappto}
3069 \providerobustcmd*\xpreappto{\global\epreappto}

\cspreappto
\csepreatto
\csgpreappto 3070 \providerobustcmd*\cspreappto[1]{%
3071   \csname itf@\ifcsname#1\endcsname
3072     preappto\else defpreapp\fi\expandafter\endcsname
3073       \csname#1\endcsname
3074 }% \cspreappto
3075 \long\def\itf@preappto#1#2#3{\edef#1{\#2\expandonce#1#3}%
3076 \long\def\itf@defpreapp#1#2#3{\edef#1{\#2#3}%
3077 \providerobustcmd*\csepreatto[1]{%
3078   \csname itf@\ifcsname#1\endcsname
3079     preappto\else defpreapp\fi\expandafter\endcsname
3080       \csname#1\endcsname
3081 }% \csepreatto
3082 \providerobustcmd*\csgpreappto{\global\cspreappto}
3083 \providerobustcmd*\csxpreappto{\global\csepreatto}



---


\removefrom
\eremovefrom
\gremovefrom 3084 \providerobustcmd\removefrom[2]{\in@false
3085   \long\def\itf@tp##1\itf@tp{%
3086     \in@true
3087     \edef#1{\unexpanded{\##1}}%
3088   }\expandafter\itf@removefrom\expandafter{\#1}{\#2}%
3089 }% \removefrom
3090 \providerobustcmd\eremovefrom[2]{\begingroup
3091   \edef\x{\endgroup\removefrom\noexpand#1{\#2}%
3092   }\x
3093 }% \eremovefrom
3094 \providerobustcmd\gremovefrom[2]{%
3095   \removefrom{\#1}{\#2}%
3096   \global\let#1=#1%
3097 }% \gremovefrom
3098 \providerobustcmd\xremovefrom[2]{%
3099   \eremovefrom{\#1}{\#2}%
3100   \global\let#1=#1%
3101 }% \xremovefrom

\itf@removefrom

```

```

3102 \long\def\itf@removefrom#1#2{%
  removes #2 (code) from #1 (hook)
3103   \def\in@@##1##2\in@@{\itf@tp##1##2\in@@@}%
3104   \def\in@@@##1\in@@@##2\in@@@{%
3105     \ifx\in@@#2\else##1\fi
3106   }\expandafter\in@@@\in@@#1\itf@tp\in@@@#2\in@@\in@@
3107 }% \itf@removefrom

```

```

\removefromstart
\eremovefromstart
\gremovefromstart
\xremovefromstart
3108 \newrobustcmd*\removefromstart[2]{\in@false
3109   \def\itf@tp##1\itf@tp{%
3110     \in@true
3111     \edef#1{\unexpanded{##1}}%
3112   }\expandafter\itf@removefromstart\expandafter{#1}{#2}%
3113 }% \removefromstart
3114 \providerobustcmd*\eremovefromstart[2]{\begingroup
3115   \edef\x{\endgroup\removefromstart\noexpand#1{#2}%
3116   }\x
3117 }% \eremovefromstart
3118 \providerobustcmd*\gremovefromstart[2]{%
3119   \removefromstart{#1}{#2}%
3120   \global\let#1=#1%
3121 }% \gremovefromstart
3122 \providerobustcmd*\xremovefromstart[2]{%
3123   \eremovefromstart{#1}{#2}%
3124   \global\let#1=#1%
3125 }% \xremovefromstart

```

```

\itf@removefromstart
3126 \def\itf@removefromstart#1#2{%
  removes #2 (code) from #1 (hook)
3127   \def\in@@##1##2\in@@{\##1\in@\itf@tp##2\in@@@}%
3128   \def\in@@@##1\in@@@##2\in@@@{%
3129     \ifx\in@@#2\else##1\fi
3130   }\expandafter\in@@@\in@@#1\itf@tp\in@@@#2\in@@
3131 }% \itf@removefromstart

```

---

3132 </etoolbox>

## 22 History

### [2010/11/16 v2.2]

- Fix a bug in `interfaces-tocloft` (key `section/dotsep`).

### [2010/11/15 v2.1]

- Improved default and blank values for bookmark levels, based on `\ttll@{section-name}`
- Correction of a bug in `\setlist`: missed a comma
- Correction of a bug in `interfaces-fancyhdr` (fixed in `./fancy@hf` handler).

### [2010/10/31 v2.0]

- Comptability of `interfaces-pgfkeys` with pgf version 2.1
- Modification of the key depth for `\tocsetup` to be able to change the t.o.c. depth anywhere inside the document (just like bookmarks depths).
- handler `./defcounter` has been replaced by more powerful `./store in counter cs`

### [2010/10/27 v1.2]

- `interfaces-hypbmsec` needed some corrections.
- The `offset` keys for fancyhdr didn't work. Fixed.
- Correction of conversion `pt` to `mu` (catcodes...)
- Added microtype to the documentation...

### [2010/10/20 v1.0]

- First version.

## 23 References

[1] *The docstrip program*; 2009/09/25 v2.5d; CTAN:[macros/latex/base/](#).

[2] *some packages...*

## Appendix: README (for CTAN)

The `interfaces` package provides a small number of macros to modify the settings of a large number of parameters provided by Most Frequently Used Package (MFUP) for typesetting beautiful document with L^AT_EX. `interfaces` is organised as different modules which are loaded after the master package is loaded. Each module can be loaded separately.

`\sectionformat`: interface for package `titlesec` to modifies the titles produced by the sectionning commands (and their bookmarks).

`\pagesetup`: interface for package `fancyhdr` to modify the current page style or create new page styles.

`\tocsetup`: interface for package `tocloft` to fine tune the aspect of the table of contents, list of figures / of tables. `\tocsetup` also applies to `\shorttableofcontents` provided by the `shorttoc` package.

`\repeatcell`: to create ‘dynamic’ tabulars, possibly with formula computation (`\numexpr`, `\pgfmathparse` or `\FPeval` formulas can be plugged in the tabular or array).

`\rowfont`: to modify the font and/or the alignment of each cell in a tabular/array row.

+ some macros to make easier the use of ε-T_EX `\marks` registers (useful with `\sectionformat` and `\pagesetup`).

**interfaces** is mainly based on `scrlfile.sty` (provided in the Koma-Script bundle, but Koma-Script classes are not required), on `pgfkeys` and `etoolbox`.

**interfaces** defines some `pgfkeys` handlers designed specially to make key-value interface for other packages. Using those handlers allows to make an interface in a few minutes...

## 24 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; plain numbers refer to the code lines where the entry is used.

| Symbols                              |                                                                                                                                   |                                                               |
|--------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------|
| \# . . . . .                         | 103, 111, 120, 689, 701                                                                                                           | \BookmarkAtEnd . . . . . 894                                  |
| \& . . . . .                         | 3059                                                                                                                              | \bookmarkget . . . . . 852, 854, 862, 864                     |
| \@input . . . . .                    | 2447                                                                                                                              | \bookmarksetup . . . . . 6, 719, 721, 723, 811, 830, 857, 867 |
| \@addtoreset . . . . .               | 540, 548, 586, 588                                                                                                                | \botmarks . . . . . 1447, 1466, 1468                          |
| \@break@tfor . . . . .               | 193                                                                                                                               | \bottomtitles . . . . . 5, 729                                |
| \@cftmakeloftitle_(patch) . . . . .  | 1763                                                                                                                              | \bottomtitlespace . . . . . 739                               |
| \@cftmakelottitle_(patch) . . . . .  | 1763                                                                                                                              | \BS@BKM@FontStyle . . . . . 807, 819, 827, 837, 850, 860      |
| \@cftmaketoctitle_(patch) . . . . .  | 1763                                                                                                                              |                                                               |
| \@cftpnumoff . . . . .               | 1619, 1745                                                                                                                        |                                                               |
| \@cftpnumon . . . . .                | 1618, 1744                                                                                                                        |                                                               |
| \@clsextension . . . . .             | 2418                                                                                                                              |                                                               |
| \@defaultunits . . . . .             | 147, 148, 199, 214, 736, 2988, 2994, 3000, 3018                                                                                   |                                                               |
| \@elt . . . . .                      | 1005                                                                                                                              |                                                               |
| \@expandtwoargs . . . . .            | 193, 208, 209                                                                                                                     |                                                               |
| \@filef@und . . . . .                | 2447                                                                                                                              |                                                               |
| \@filelist . . . . .                 | 2372, 2377                                                                                                                        |                                                               |
| \@firstofone . . . . .               | 1866, 2141                                                                                                                        |                                                               |
| \@fourthoffive . . . . .             | 929                                                                                                                               |                                                               |
| \@ifclassloaded . . . . .            | 2422                                                                                                                              |                                                               |
| \@ifpackageloaded . . . . .          | 2421                                                                                                                              |                                                               |
| \@input_(patch) . . . . .            | 2445                                                                                                                              |                                                               |
| \@mainaux . . . . .                  | 774, 888, 893                                                                                                                     |                                                               |
| \@nobreakfalse . . . . .             | 994                                                                                                                               |                                                               |
| \@onelevel@sanitize . . . . .        | 2027                                                                                                                              |                                                               |
| \@onlypreamble . . . . .             | 52, 2366                                                                                                                          |                                                               |
| \@pkgextension . . . . .             | 2418                                                                                                                              |                                                               |
| \@tfor . . . . .                     | 191, 2591, 2630, 2644,<br>2657, 2674, 2740, 2759, 2774, 2796, 2814, 2833,<br>2853, 2871, 2891, 2908, 2931, 2950, 2968, 2991, 3015 |                                                               |
| \@whilenum . . . . .                 | 1876, 1885                                                                                                                        |                                                               |
| \\\bookmark_(patch) . . . . .        | 773                                                                                                                               |                                                               |
| \~ . . . . .                         | 1857                                                                                                                              |                                                               |
| A                                    |                                                                                                                                   |                                                               |
| \active . . . . .                    | 1830, 1861                                                                                                                        |                                                               |
| \addcontentsline_(patch) . . . . .   | 688                                                                                                                               |                                                               |
| \AddLineBeginMainAux . . . . .       | 420                                                                                                                               |                                                               |
| \addtocontents . . . . .             | 1641, 1671, 1675                                                                                                                  |                                                               |
| \adl@arrayrestore . . . . .          | 76, 83                                                                                                                            |                                                               |
| \adl@endarray . . . . .              | 76, 83                                                                                                                            |                                                               |
| \AfterFile . . . . .                 | 427, 1611, 2429                                                                                                                   |                                                               |
| \aftergroup . . . . .                | 843, 884, 1820                                                                                                                    |                                                               |
| \AfterPackage . . . . .              | 55,                                                                                                                               |                                                               |
|                                      | 68, 72, 89, 246, 287, 684, 940, 2140, 2393, 2409                                                                                  |                                                               |
| \allpgfmodules . . . . .             | 298, 386, 403                                                                                                                     |                                                               |
| \alltikzlibraries . . . . .          | 308, 387                                                                                                                          |                                                               |
| \alltikzpackages . . . . .           | 293, 382, 385, 400                                                                                                                |                                                               |
| \appendices . . . . .                | 2324                                                                                                                              |                                                               |
| \apptocmd . . . . .                  | 98, 118, 1784                                                                                                                     |                                                               |
| B                                    |                                                                                                                                   |                                                               |
| \BeforeFile . . . . .                | 423, 1607, 2429                                                                                                                   |                                                               |
| \BeforePackage . . . . .             | 38, 2403                                                                                                                          |                                                               |
| \BKM@ref . . . . .                   | 931                                                                                                                               |                                                               |
| \BKM@color . . . . .                 | 805, 815, 824, 836                                                                                                                |                                                               |
| \BKM@dest . . . . .                  | 927, 929                                                                                                                          |                                                               |
| \BKM@page . . . . .                  | 911                                                                                                                               |                                                               |
| \bookmark . . . . .                  | 6, 773, 776, 779, 898, 915, 1805                                                                                                  |                                                               |
| C                                    |                                                                                                                                   |                                                               |
| \C . . . . .                         | 187, 1887, 1894, 1968, 2089                                                                                                       |                                                               |
| \c . . . . .                         | 1887, 1889, 2092                                                                                                                  |                                                               |
| \c@lofdepth . . . . .                | 1671                                                                                                                              |                                                               |
| \c@lotdepth . . . . .                | 1675                                                                                                                              |                                                               |
| \c@tocdepth . . . . .                | 1640, 1641                                                                                                                        |                                                               |
| \cftdot . . . . .                    | 1525                                                                                                                              |                                                               |
| \cftdotfill . . . . .                | 1727                                                                                                                              |                                                               |
| \cftdotsep . . . . .                 | 1522, 1730                                                                                                                        |                                                               |
| \cftnodots . . . . .                 | 1524, 1735, 1736                                                                                                                  |                                                               |
| \changefont . . . . .                | 3, 243, 271                                                                                                                       |                                                               |
| \chapter . . . . .                   | 1020, 1690, 2347                                                                                                                  |                                                               |
| \color . . . . .                     | 232, 233, 274, 278,<br>1090, 1110, 1208, 1217, 1226, 1235, 1244, 1253,<br>1262, 1271, 1280, 1289, 1298, 1307, 2686, 2710, 2712    |                                                               |
| \colorlet . . . . .                  | 229                                                                                                                               |                                                               |
| \column . . . . .                    | 1887, 1968, 1999                                                                                                                  |                                                               |
| \columnnsep . . . . .                | 1542, 1571                                                                                                                        |                                                               |
| \columnseprule . . . . .             | 1541, 1566                                                                                                                        |                                                               |
| \columnseprulecolor . . . . .        | 1543                                                                                                                              |                                                               |
| \contentsname . . . . .              | 1600, 1645, 1646, 1764, 1820                                                                                                      |                                                               |
| \count . . . . .                     | 147                                                                                                                               |                                                               |
| \countdef . . . . .                  | 1868, 1869, 1870                                                                                                                  |                                                               |
| \countvalues . . . . .               | 11, 1832                                                                                                                          |                                                               |
| \csepreatpto . . . . .               | 2500, 3070                                                                                                                        |                                                               |
| \csgappto . . . . .                  | 2156, 2428                                                                                                                        |                                                               |
| \csgpreappto . . . . .               | 3070                                                                                                                              |                                                               |
| \cspreappto . . . . .                | 3070                                                                                                                              |                                                               |
| \cxspreatpto . . . . .               | 3070                                                                                                                              |                                                               |
| \CT@cell@color . . . . .             | 2142, 2143                                                                                                                        |                                                               |
| \CT@cellc . . . . .                  | 2140                                                                                                                              |                                                               |
| \CT@color . . . . .                  | 2142                                                                                                                              |                                                               |
| \CT@end . . . . .                    | 70, 76                                                                                                                            |                                                               |
| \currentgrouplevel . . . . .         | 809, 842                                                                                                                          |                                                               |
| D                                    |                                                                                                                                   |                                                               |
| \default@color . . . . .             | 233                                                                                                                               |                                                               |
| \DefineVerbatimEnvironment . . . . . | 2291, 2293, 2295                                                                                                                  |                                                               |
| \diaghead . . . . .                  | 2135                                                                                                                              |                                                               |
| \dimen . . . . .                     | 2214, 2215,<br>2216, 2223, 2232, 2234, 2237, 2238, 2243, 2251, 2254                                                               |                                                               |
| \dimendef . . . . .                  | 2225, 2226, 2227                                                                                                                  |                                                               |
| \dump . . . . .                      | 115                                                                                                                               |                                                               |
| E                                    |                                                                                                                                   |                                                               |
| \embedfile . . . . .                 | 2367, 2386                                                                                                                        |                                                               |
| \embedthisfile . . . . .             | 17, 2362, 2365, 2366                                                                                                              |                                                               |
| \endarray . . . . .                  | 68                                                                                                                                |                                                               |
| \endlinechar . . . . .               | 425, 429                                                                                                                          |                                                               |
| \preappto . . . . .                  | 3062                                                                                                                              |                                                               |
| \removefrom . . . . .                | 3084                                                                                                                              |                                                               |
| \removefromstart . . . . .           | 3108                                                                                                                              |                                                               |

|                                               |                                                      |
|-----------------------------------------------|------------------------------------------------------|
| <b>F</b>                                      |                                                      |
| \f@baselineskip . . . . .                     | 258, 261, 266                                        |
| \f@linespread . . . . .                       | 258, 261, 264, 265, 266, 267, 268, 269               |
| \f@size . . . . .                             | 258, 261, 266                                        |
| \fancy@setoffs . . . . .                      | 1367, 1371, 1376, 1380, 1384, 1388, 1392, 1396       |
| \fancyhf . . . . .                            | 1032                                                 |
| \fancyhffoffset . . . . .                     | 1033                                                 |
| \figure . . . . .                             | 1696                                                 |
| \firstmarks . . . . .                         | 1442, 1455, 1465, 1468                               |
| \fontdimen . . . . .                          | 175, 181                                             |
| \fontencoding . . . . .                       | 249                                                  |
| \fontfamily . . . . .                         | 251                                                  |
| \fontseries . . . . .                         | 253                                                  |
| \fontshape . . . . .                          | 255                                                  |
| \footrule . . . . .                           | 1112, 1113, 1114, 1115                               |
| \footrulewidth . . . . .                      | 1118, 1119, 1120                                     |
| \FPeval . . . . .                             | 2021, 2050, 2071, 2072                               |
| \FPresult . . . . .                           | 2072, 2073, 2075                                     |
| \futurelet . . . . .                          | 103, 158, 2697, 2704, 2720, 2728                     |
| <b>G</b>                                      |                                                      |
| \getrefbykeydefault . . . . .                 | 911                                                  |
| \getthebotmarks . . . . .                     | 1465                                                 |
| \getthefirstmarks . . . . .                   | 1465                                                 |
| \getthemarks . . . . .                        | 10, 1455, 1465, 1466, 1467, 1471, 1472               |
| \gettethopmarks . . . . .                     | 1465                                                 |
| \gluetomu . . . . .                           | 172                                                  |
| \gpreappto . . . . .                          | 3062                                                 |
| \gremovefrom . . . . .                        | 3084                                                 |
| \gremovefromstart . . . . .                   | 3108                                                 |
| <b>H</b>                                      |                                                      |
| \hbs@do_(patch) . . . . .                     | 947                                                  |
| \hbs@process_(patch) . . . . .                | 957                                                  |
| \headrule . . . . .                           | 1092, 1093, 1094, 1095                               |
| \headrulewidth . . . . .                      | 1098, 1099, 1100                                     |
| \headwidth . . . . .                          | 1099, 1120, 1403, 1404, 1405, 1408, 1409, 1412       |
| \htruncadjust . . . . .                       | 16, 2200                                             |
| \htruncwidth . . . . .                        | 2266                                                 |
| \HyperRaiseLinkHook . . . . .                 | 750, 751, 752, 753                                   |
| \HyperRaiseLinkLength . . . . .               | 754                                                  |
| <b>I</b>                                      |                                                      |
| \if@nobreak . . . . .                         | 988                                                  |
| \ifcsundef . . . . .                          | 460,                                                 |
|                                               | 463, 499, 546, 553, 581, 1057, 1061, 1429, 2174      |
| \ifcsvoid . . . . .                           | 639, 640                                             |
| \ifitf@celltranspose . . . . .                | 1854, 1898, 2115                                     |
| \ifmarksequal . . . . .                       | 10, 1468                                             |
| \ifnotempty . . . . .                         | 2405                                                 |
| \ifttl@explicit . . . . .                     | 508                                                  |
| \ifundef . . . . .                            | 1020, 2347, 2394                                     |
| \ignorespaces . . . . .                       | 1945, 2193, 2213, 2233                               |
| \in@ . . . . .                                | 193, 208, 209, 400, 403, 3105, 3106, 3127, 3129      |
| \interitem . . . . .                          | 15, 2152                                             |
| \itf@AfterPackage . . . . .                   | 65                                                   |
| \itf@ampactive . . . . .                      | 1855, 1858, 1917, 1939                               |
| \itf@BKM@enable . . . . .                     | 783,                                                 |
|                                               | 785, 786, 787, 788, 789, 790, 885, 886, 890, 892     |
| \itf@BKM@pageref . . . . .                    | 898                                                  |
| \itf@BKM@ref . . . . .                        | 915                                                  |
| \itf@BKM@restore . . . . .                    | 843, 871, 875, 884                                   |
| \itf@BKM@write . . . . .                      | 774, 883                                             |
| \itf@cell@calc . . . . .                      | 2056                                                 |
| <b>J</b>                                      |                                                      |
| \itf@cell@formulaerror . . . . .              | 1983, 1984, 1986, 2009, 2018, 2021, 2026, 2055       |
| \itf@cell@FPcalc . . . . .                    | 2022, 2056                                           |
| \itf@cell@getformula . . . . .                | 1988, 2006                                           |
| \itf@cell@init . . . . .                      | 1877,                                                |
|                                               | 1880, 1886, 1887, 1891, 1893, 1897, 1917, 1939       |
| \itf@cell@insert . . . . .                    | 1918, 1940, 2077                                     |
| \itf@cell@numcalc . . . . .                   | 1999, 2000, 2015, 2056                               |
| \itf@cell@pgfcalc . . . . .                   | 2019, 2056                                           |
| \itf@celltransposefalse . . . . .             | 1993                                                 |
| \itf@cft@label . . . . .                      | 1791                                                 |
| \itf@cft@newlistof . . . . .                  | 1500, 1633, 1634, 1635, 1785                         |
| \itf@cft@newsection . . . . .                 | 570, 1709, 1751, 1754, 1756, 1759, 1761              |
| \itf@cft@secname . . . . .                    | 1685, 1686, 1688, 1744, 1745                         |
| \itf@cft@sectionname . . . . .                | 1685                                                 |
| \itf@countvalues . . . . .                    | 1833, 1835, 1839, 1841                               |
| \itf@diaghead . . . . .                       | 2116, 2120, 2132                                     |
| \itf@expandonceifcs . . . . .                 | 137, 1976, 2004                                      |
| \itf@getthemarks . . . . .                    | 1458, 1462, 1464                                     |
| \itf@getvalue . . . . .                       | 1842, 1882, 1895                                     |
| \itf@hbs@chkclass . . . . .                   | 981, 986, 989                                        |
| \itf@hbs@n@breakfix . . . . .                 | 990, 991                                             |
| \itf@hbs@next . . . . .                       | 958, 967                                             |
| \itf@hbs@nobreakfix . . . . .                 | 988, 990, 996                                        |
| \itf@htruncadjust . . . . .                   | 2201, 2203, 2261                                     |
| \itf@htruncwidth . . . . .                    | 2267, 2269, 2274, 2276                               |
| \itf@last@from@next . . . . .                 | 603, 984, 997                                        |
| \itf@makecell . . . . .                       | 1866, 1867, 2128                                     |
| \itf@marks@new . . . . .                      | 1421, 1427, 1435                                     |
| \itf@message . . . . .                        | 1943, 1947                                           |
| \itf@musk . . . . .                           | 129, 1521, 1522, 1732, 1733                          |
| \itf@number . . . . .                         | 146, 1954, 1973, 2001, 2164                          |
| \itf@pagesetup . . . . .                      | 1037, 1038, 1040, 1042                               |
| \itf@patch . . . . .                          | 103, 691, 773, 947, 957, 1763, 1770, 1777, 1784      |
| \itf@pgfk@color . . . . .                     | 2685, 2693, 2697, 2704                               |
| \itf@pgfk@colorcmd . . . . .                  | 2708, 2716, 2720, 2728                               |
| \itf@pgfkeys@aliaserror . . . . .             | 2579, 2581, 2587                                     |
| \itf@pgfkeys@case@three . . . . .             | 2459, 2465, 2554, 2563, 2565                         |
| \itf@pgfkeys@case@handleall . . . . .         | 2460, 2565                                           |
| \itf@pgfkeys@default@blank@required . . . . . | 2514, 2524, 2529, 2558                               |
| \itf@pgfkeys@ifaliasexists . . . . .          | 2569, 2575, 2580, 2589                               |
| \itf@preappto . . . . .                       | 3075                                                 |
| \itf@preaptocmd . . . . .                     | 96, 106, 691                                         |
| \itf@rem@mu . . . . .                         | 165, 169                                             |
| \itf@removefrom . . . . .                     | 3088, 3102                                           |
| \itf@removefromstart . . . . .                | 3112, 3126                                           |
| \itf@repeatcell . . . . .                     | 1862, 1864, 1946                                     |
| \itf@resetlastsections . . . . .              | 1005, 1008, 1013                                     |
| \itf@scaledimen . . . . .                     | 2236, 2251, 2263, 2265                               |
| \itf@scalefactor . . . . .                    | 2262, 2264                                           |
| \itf@sectionformat . . . . .                  | 437, 438, 440                                        |
| \itf@setbottomtitlespace . . . . .            | 732, 735, 738                                        |
| \itf@setlist . . . . .                        | 2156, 2157, 2159, 2180                               |
| \itf@setlist@error . . . . .                  | 2175, 2181, 2186                                     |
| \itf@sk . . . . .                             | 128, 733, 736, 739,                                  |
|                                               | 2988, 2989, 2994, 2996, 2998, 3001, 3003, 3018, 3024 |
| \itf@strip@at . . . . .                       | 132, 486, 948                                        |
| \itf@strip@dot . . . . .                      | 132, 174, 180                                        |
| \itf@strip@mu . . . . .                       | 164, 1522, 1733                                      |
| \itf@tk . . . . .                             | 130, 1401                                            |
| \itf@tocloft . . . . .                        | 1623, 1764, 1771, 1778, 1791, 1795                   |
| \itf@ttl@BKM . . . . .                        | 692, 702, 728                                        |
| \itf@ttl@createsection . . . . .              | 559, 582                                             |

|                                      |                                                                    |                                             |                                           |  |  |
|--------------------------------------|--------------------------------------------------------------------|---------------------------------------------|-------------------------------------------|--|--|
| \itf@ttl@getformat . . . . .         | 451                                                                | \pagestyle . . . . .                        | 1049, 1057, 1080, 1410                    |  |  |
| \itf@ttl@getspacing . . . . .        | 451                                                                | \paragraph . . . . .                        | 1694                                      |  |  |
| \itf@ttl@newsection . . . . .        | 459, 578                                                           | \patchcmd . . . . .                         | 105, 127, 773                             |  |  |
| \itf@ttl@sectionformat . . . . .     | 450                                                                | \pdfstringdefDisableCommands . . . . .      | 247                                       |  |  |
| \itf@ttl@setkey . . . . .            | 467,<br>471, 473, 474, 475, 476, 477, 488, 489, 490, 491, 492      | \pgfkeys@case@three . . . . .               | 2459, 2465, 2553                          |  |  |
| \itf@ttl@setup . . . . .             | 451                                                                | \pgfkeys@case@three@handleall . . . . .     | 2460                                      |  |  |
| \itf@ttl@titleclass . . . . .        | 526, 586, 588                                                      | \pgfkeys@exp@call . . . . .                 | 2467, 2510                                |  |  |
| \itf@ttl@titleclass@i . . . . .      | 529, 530, 532, 544                                                 | \pgfkeys@global@temp . . . . .              | 2485, 2492, 2501, 2508                    |  |  |
| \itf@ttl@titleclass@ii . . . . .     | 540, 545, 558                                                      | \pgfkeys@ifexecutehandler . . . . .         | 2456, 2483, 2495, 2499, 2511, 2515, 2564  |  |  |
| \itf@ttl@update . . . . .            | 454, 494                                                           | \pgfkeys@unpack . . . . .                   | 2458, 2464                                |  |  |
| \itf@undefinedpagestyle . . . . .    | 1058, 1062, 1065, 1068                                             | \pgfkeysaddvalue_(bug_correction) . . . . . | 2483                                      |  |  |
| \itf@units@mu . . . . .              | 184, 1521, 1732                                                    | \pgfkeysEaddvalue . . . . .                 | 1595, 1601, 1989, 2495                    |  |  |
| \itf@units@mutopt . . . . .          | 171, 212                                                           | \pgfkeysEsetvalue . . . . .                 | 2495                                      |  |  |
| \itf@units@normalize . . . . .       | 225                                                                | \pgfkeysmeaning . . . . .                   | 2474                                      |  |  |
| \itf@units@pt . . . . .              | 203                                                                | \pgfkeysnovalue . . . . .                   | 1511, 2539, 2573, 2682, 2690, 2698, 2705  |  |  |
| \itf@units@pttomu . . . . .          | 171, 197                                                           | \pgfkeysvalueof@unexpanded . . . . .        | 2479, 2633, 2647, 2743, 2836, 2874        |  |  |
| \itf@units@scale . . . . .           | 217, 2838,<br>2857, 2876, 2895, 2917, 2937, 2955, 2976, 2998, 3023 | \pgfkeysvaluerequired . . . . .             | 2544                                      |  |  |
| \itf@units@scale@normalize . . . . . | 219, 220                                                           | \pgfmathparse . . . . .                     | 2018, 2046, 2066                          |  |  |
| \itf@usetikzcode . . . . .           | 389, 398                                                           | \pgfmathresult . . . . .                    | 2067                                      |  |  |
| <b>L</b>                             |                                                                    |                                             |                                           |  |  |
| \label . . . . .                     | 1816                                                               | \preappto . . . . .                         | 3062                                      |  |  |
| \lastchapterbookmark . . . . .       | 1023                                                               | \pretocmd . . . . .                         | 97, 118, 947, 957, 1622, 1763, 1770, 1777 |  |  |
| \lastchaptertitle . . . . .          | 1021                                                               | <b>R</b>                                    |                                           |  |  |
| \lastchaptertoc . . . . .            | 1022                                                               | \R . . . . .                                | 1877, 1881, 2084                          |  |  |
| \lastsection . . . . .               | 955, 965                                                           | \r . . . . .                                | 1877, 1879, 2087                          |  |  |
| \lastsectionname . . . . .           | 992, 993, 998, 1014                                                | \raisedhyperdef . . . . .                   | 6, 746, 1801                              |  |  |
| \listfigurename . . . . .            | 1657, 1658, 1771                                                   | \rem@pt . . . . .                           | 265                                       |  |  |
| \listofsetup . . . . .               | 9, 1488, 1630, 1790                                                | \remove@to@nnil . . . . .                   | 142, 1837, 1847, 1849, 2024               |  |  |
| \listtablename . . . . .             | 1663, 1664, 1778                                                   | \removefrom . . . . .                       | 22, 3084                                  |  |  |
| \LoadInterface . . . . .             | 6, 7, 9, 40                                                        | \removefromstart . . . . .                  | 1949, 1951, 3108                          |  |  |
| \lofsetup . . . . .                  | 1488, 1776                                                         | \ReNewEnviron . . . . .                     | 2394, 2395                                |  |  |
| \lotsetup . . . . .                  | 1488, 1783                                                         | \RenewEnviron . . . . .                     | 2395                                      |  |  |
| \ltx@clsextension . . . . .          | 2420                                                               | \repeatcell . . . . .                       | 11, 1854, 2005, 2037, 2040                |  |  |
| \ltx@ifclassloaded . . . . .         | 2422                                                               | \row . . . . .                              | 1877, 2000                                |  |  |
| \ltx@ifpackageloaded . . . . .       | 53, 57, 66, 73, 90, 227, 231, 1532, 2298, 2421                     | <b>S</b>                                    |                                           |  |  |
| \ltx@pkgextension . . . . .          | 2419                                                               | \scr@load@hook . . . . .                    | 2424, 2446, 2448                          |  |  |
| <b>M</b>                             |                                                                    |                                             |                                           |  |  |
| \makecell . . . . .                  | 1867                                                               | \scr@set@load@hook_(optimisation) . . . . . | 2428                                      |  |  |
| \marks . . . . .                     | 43, 1423, 1431                                                     | \sectionformat . . . . .                    | 4, 436, 721                               |  |  |
| \marksthe . . . . .                  | 10, 633, 1419                                                      | \selectfont . . . . .                       | 244                                       |  |  |
| \marksthecs . . . . .                | 1425                                                               | \set@fontsize . . . . .                     | 258, 261, 266                             |  |  |
| \maxdimen . . . . .                  | 2192                                                               | \setlist . . . . .                          | 15                                        |  |  |
| \muexpr . . . . .                    | 179                                                                | \setlist_(extension,_star_form) . . . . .   | 2155                                      |  |  |
| \multicolssep . . . . .              | 1539                                                               | \skip . . . . .                             | 171, 184, 203                             |  |  |
| \muskip . . . . .                    | 171, 184, 203                                                      | \strip@pt . . . . .                         | 2262                                      |  |  |
| \mutoglu . . . . .                   | 179                                                                | \strut . . . . .                            | 1076                                      |  |  |
| <b>N</b>                             |                                                                    |                                             |                                           |  |  |
| \newcounter . . . . .                | 547                                                                | \subparagraph . . . . .                     | 1695                                      |  |  |
| \newlist . . . . .                   | 2184                                                               | \subsubsection . . . . .                    | 1697                                      |  |  |
| \newlistentry_(patch) . . . . .      | 1763                                                               | \subsubsection . . . . .                    | 1693                                      |  |  |
| \newmarks . . . . .                  | 1436                                                               | \subtable . . . . .                         | 1699                                      |  |  |
| \newmuskip . . . . .                 | 129                                                                | <b>T</b>                                    |                                           |  |  |
| \newskip . . . . .                   | 128                                                                | \tab . . . . .                              | 1856, 1860                                |  |  |
| \newtoks . . . . .                   | 130                                                                | \table . . . . .                            | 1698                                      |  |  |
| \normalbaselineskip . . . . .        | 1141                                                               | \textfont . . . . .                         | 175, 181                                  |  |  |
| \normalcolor . . . . .               | 1576, 2688, 2714, 2723, 2731                                       | \textheight . . . . .                       | 736                                       |  |  |
| <b>P</b>                             |                                                                    |                                             |                                           |  |  |
| \pagesetup . . . . .                 | 9, 1034, 1078, 1401                                                | \the@dimen . . . . .                        | 2207, 2245, 2254                          |  |  |
| <b> </b>                             |                                                                    |                                             |                                           |  |  |
| \tikzset . . . . .                   | 289                                                                | \thebotmarks . . . . .                      | 1440                                      |  |  |

|                    |                              |                                     |
|--------------------|------------------------------|-------------------------------------|
| \titleclass        | 419, 534, 537, 562           | <b>U</b>                            |
| \titleformat       | 419                          | \unless ..... 992, 993              |
| \titlespacing      | 419                          | \unskip ..... 2193, 2213, 2233      |
| \tocloftpagestyle  | 1502                         | \usepgfmodule ..... 386, 404        |
| \tocsetup          | 8, 8, 1488, 1769             | \usetikz ..... 3, 390, 391, 2046    |
| \toks              | 2215, 2221, 2236, 2240, 2251 | \usetikzlibrary ..... 376, 387, 405 |
| \toksdef           | 1871                         | <b>V</b>                            |
| \topmarks          | 1452, 1467                   | \vsize ..... 2192                   |
| \truncate          | 2195                         | \vskip ..... 1100, 1119, 1120       |
| \TruncateMarker    | 2190                         | <b>W</b>                            |
| \trunc{text}       | 16, 2190                     | \write ..... 774, 888, 893          |
| \ttl@getkeys       | 496, 497                     | <b>X</b>                            |
| \ttl@spacing@ii    | 519                          | \xpreappto ..... 3062               |
| \ttlh@runinifempty | 431                          | \xremovefrom ..... 3084             |
| \ttlp@append       | 415, 421                     | \xremovefromstart ..... 3108        |