

# The **interfaces**\* package

## interfaces with keys for other packages

FC

2010/11/21 – version 2.3  
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**<sup>1</sup> and **scrfile** for its different modules are loaded `\AfterPackage`.  $\epsilon$ -**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}` *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-scrfile**, **interfaces-etoolbox**<sup>2</sup> and **interfaces-pgfkeys**: those are the *system-layer* of **interfaces**.

### ~~~~~ Example ~~~~~

```
\pagesetup[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
}
\pagesetup[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-hyphbmsec
  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},
}

\sectionformat\MiniSection{% new sectionning command
  class=straight[\subparagraph],
  shape=block,
  label=,
  top=3pt plus2pt, bottom=1pt,
  font=\bfseries\sffamily,
  bookmark=off,% or false or none
  break=\allowbreak,
  mark=\marksthe{MiniSection}{#1},% see interfaces-marks
}
\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,
  subsection/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](http://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

<b>1</b>	<b>Interface L<sup>A</sup>T<sub>E</sub>X</b>	<b>3</b>	<b>21</b>	<b>Implementation</b>	<b>25</b>
<b>2</b>	<b>Interface TikZ</b>	<b>3</b>	21.1	Interfaces	25
<b>3</b>	<b>Interface titlesec: \sectionformat</b>	<b>4</b>	21.1.1	Requirements	
<b>4</b>	<b>Interface hyperref</b>	<b>6</b>	21.1.2	The package options / list of interfaces	
<b>5</b>	<b>Interface bookmark</b>	<b>6</b>	21.2	Interfaces-base	25
<b>6</b>	<b>Interface hypbmsec</b>	<b>7</b>	21.2.1	Bug Fixes	
<b>7</b>	<b>Interface tocloft: \tocsetup</b>	<b>8</b>	21.2.2	Interfaces patches	
<b>8</b>	<b>Interface fancyhdr: \pagesetup</b>	<b>9</b>	21.2.3	Helper macros	
<b>9</b>	<b>Interface marks</b>	<b>10</b>	21.2.4	Numbers	
<b>10</b>	<b>Interface makecell: \repeatcell</b>	<b>11</b>	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 <sup>A</sup> T <sub>E</sub> X	30
10.3	Empty rows	12	21.3.1	\change font	
10.4	color palette	13	21.4	Interface TikZ	31
10.5	tabularcalc-like tables (with dcolumn package)		21.4.1	Catcode verifications	
	and pgfmath	13	21.4.2	List of pgf/TikZ packages and libraries	
10.6	Double entry array: the Body Mass Index	14	21.4.3	pgfkeys definitions for \usetikz	
<b>11</b>	<b>Interface enumitem</b>	<b>15</b>	21.5	Interface titlesec: \sectionformat	34
<b>12</b>	<b>Interface truncate</b>	<b>16</b>	21.5.1	titlesec modifications	
<b>13</b>	<b>Interface appendix</b>	<b>17</b>	21.5.2	\sectionformat	
<b>14</b>	<b>Interface embedfile: \embedthisfile</b>	<b>17</b>	21.5.3	Keys for \sectionformat	
<b>15</b>	<b>Interface environ</b>	<b>17</b>	21.5.4	\bottomtitles	
<b>16</b>	<b>Interface umrand</b>	<b>17</b>	21.6	Interface hyperref	40
<b>17</b>	<b>Interface pgfkeys</b>	<b>18</b>	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	54
17.10	/.store as color in cs	21	21.11	Interface tocloft: \tocsetup	56
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	62
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	68
17.17	/.estore as dimen in cs	22	21.14	Interface truncate	69
<b>18</b>	<b>Interface etoolbox</b>	<b>22</b>	21.15	Interface appendix	71
18.1	New macros provided with \providecommand	22	21.16	Interface embedfile: \embedthisfile	72
<b>19</b>	<b>Interface scrfile</b>	<b>22</b>	21.17	Interface environ	72
19.1	Star forms for \BeforeFile and \AfterFile	22	21.18	Interface umrand	72
19.2	Allowing \BeforeFile and \AfterFile when \@input is invoked	23	21.19	Interface scrfile	73
<b>20</b>	<b>List of corrected or patched commands</b>	<b>23</b>	21.20	Interface pgfkeys	74
20.1	LaTeX	23	21.20.1	interfaces-pgfkeys	
20.2	bookmark	23	21.21	Interface etoolbox	86
20.3	colortbl	24	<b>22</b>	<b>History</b>	<b>89</b>
20.4	hypbmsec	24	[2010/11/21 v2.3]		89
20.5	enumitem	24	[2010/11/16 v2.2]		89
20.6	pgfkeys	24	[2010/11/15 v2.1]		89
			[2010/10/31 v2.0]		89
			[2010/10/27 v1.2]		89
			[2010/10/20 v1.0]		89
			<b>23</b>	<b>References</b>	<b>89</b>
			<b>README</b>		<b>89</b>
			<b>24</b>	<b>Index</b>	<b>91</b>

# 1 Interface L<sup>A</sup>T<sub>E</sub>X

```
\change font { encoding | enc = changes \font encoding
                family | fam = changes \font family
                series | ser = changes \font series
                shape | sha = changes \font shape
                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%:

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

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

Finally, `\change font` 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<br>pgfmath<br>pgfcalendar<br>pgfplots<br>pgfpages<br>tikz | matrix<br>plot | arrows<br>chains<br>er<br>scopes<br>snakes<br>shapes<br>shapes.misc<br>shapes.geometric<br>shapes.arrows<br>shapes.multipart<br>shapes.callouts<br>shapes.symbols<br>shapes.gates.logic.US<br>shapes.gates.logic.IEC<br>decorations<br>decorations.pathmorphing | decorations.pathreplacing<br>decorations.shapes<br>decorations.text<br>decorations.markings<br>decorations.fractals<br>decorations.footprints<br>positioning,trees<br>automata<br>patterns<br>fadings<br>shadows<br>through<br>backgrounds<br>petri<br>fit<br>mindmap<br>folding<br>plohandlers<br>plotmarks<br>calc<br>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  $\text{\TeX}$  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 ~~~~~                         |                               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|------------------------------------------|-------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>class</code>                       | name of the class             | <code>page</code> =like standard <code>\part</code><br><code>top</code> =like standard <code>\chapter</code><br><code>straight</code> =like standard <code>\section</code> etc.<br>example: <code>\sectionformat\subchapter{class=straight[\chapter]}</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <code>shape</code>                       | shape                         | <code>hang</code> =hanging label (like <code>\section</code> )<br><code>block</code> =paragraph without formatting (for centered titles)<br><code>display</code> =like standard <code>\chapter</code><br><code>runin</code> =like standard <code>\paragraph</code><br><code>leftmargin</code> =left margin<br><code>rightmargin</code> =right margin<br><code>drop</code> =wraps the text around the title (the first paragraph should be longer than the title, otherwise, they overlap)<br><code>wrap</code> =like <code>drop</code> but the space reserved for the title is variable<br><code>frame</code> =like <code>display</code> +puts the title in frame<br><code>runinifempty</code> = <code>hang</code> but no label if the title is empty |
| <code>font</code> ≡ <code>format</code>  | format for title              | applied to the label and the title                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <code>label</code>                       | title label                   | ex: <code>\arabic{section}</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <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 <code>\sectionmark</code> : redefines <code>\...mark#1...</code> (the argument is the title)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| <code>break</code>                       | parameterless macro           | ex <code>\sectionbreak</code> to be executed just before the section command                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <code>thesection</code>                  | toc and ref labels            | redefines <code>\thesection</code><br><code>\thesection</code> , <code>\thesubsection</code> 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:

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

---

```
\sectionformat\section[hang]{%      this is the same as:  shape = hang
  font=\bfseries\scshape,
  label=\Roman{chapter}\ \textbullet\ \arabic{section},
  bottom=lex plus.4ex minus.2ex,
  top+=3pt,
  mark=\markright{#1},% => this defines \def\sectionmark#1{\markright{#1}}
  break=\goodbreak      %=> this defines \def\sectionbreak{\goodbreak}
  thesection=\Roman{chapter}.\arabic{section} } % redefines \def\thesection{...}
```

---

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

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

One swallow does not a summer make, `interface-titlesec` does not provides 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 (*ie.* `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`.

```
\bottomtitles{<glue|percentage of \textheight>}
```

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

```
\bottomtitles{.2}          set \bottomtitlespace to .2\textheight
\bottomtitles=.2\textheight is the same
```

You can set a negativ 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`:

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

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:

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

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 `\sim\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 of 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:

```
\lastsectiontitle      expands to the title of the last \section
\lastsectionbookmark    expands to the text of the bookmark for the last \section
\lastsectiontoc         expands to the entry of the last \section in the table of contents
```

Macros for each sectionning command are defined:

```
\lastparttitle          \lastchaptertitle
\lastsectiontitle       \lastsubsectiontitle      \lastsubsubsectiontitle
\lastparagraphtitle     \lastsubparagraphtitle
```

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 `\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, *ie.* if `\sectionformat\chapter` or `\titleformat\chapter` has been invoked. Otherwise, `\lastchaptertitle`, `\lastchapterbookmark` and `\lastchaptertoc` 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 `\ifmarksequal`). With this feature, when a new `\section` is created, `\lastsubsection` is reset to `\relax` therefore, `\marksthecs{<subsection>}{<lastsubsection>}` will not mark anything.

## 7 Interface `tocloft` : `\tocsetup`, `\lofsetup` and `\lotsetup`

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...

```
\tocsetup{
    depth=\setcounter{tocdepth} This is a LATEX rather than tocloft's ...
    However with interfaces you can change the t.o.c. depth anywhere
    inside your document, just like bookmarks depths (two runs are required)

    title|contentsname=redefines \contentsname
    title/top|title/top+|title/top-|title/top*=vertical space before the title (set|add|subtract|scale)
    title/bottom|title/bottom+ etc.=vertical space after the title
    title/font|title/font*=the title font (set|append code)
    title/after|title/after*=hook to execute just after the title (\cftaftertocitle...)
    parskip|parskip+ etc.|parsep=modifies (or add to) \cftparskip
    dotsep=modifies \cftdotsep if no dimen is specified, mu is assumed, otherwise, conversion
    to  $\mu \approx 1em/18$  is supplied. Default value is  $4.5\mu$  and blank value is the same as
    \cftnodots
    dot=modifies \cftdot, the dot used for leaders produced by \cftdotfill
    pagestyle=page style for the first page (\tocloftpagestyle)
    mark=the mark command: \cftmarktoc
    example: \chaptermark\contentsname

    before|before*|after|after*=hooks to execute before or after the command \tableofcontents
    twocolumns|columns|multicol=twocolumns=true is the same as columns=2 Makes a multicolumn table
    of contents. Package multicol is required!
    columns/sep|columns/sep+|columnsep etc.=sets (or add) a  $\langle dimen \rangle$  to \columnsep
    columns/rule|columns/rule+|columnseprule etc.=sets (or add) a  $\langle dimen \rangle$  to \columnseprule
    columns/color|columnseprulecolor=modifies \columnseprulecolor
    multicolsep|multicolsep+ etc.=modifies (or add) a  $\langle glue \rangle$  to \multicolsep
    \multicolsep is the space added before and after multicol environ-
    ment. Its default value is 0pt plus 4pt minus 3pt.
    multicol/begin|twocolumns/before=hooks just after \begin{multicols} ...
    multicol/end|twocolumns/after=... and before \end{multicols}

    label=the label for reference to the table of contents
    ref=the reference printed by \ref when referencing the table of contents
    (default is \contentsname. In any case, \nameref prints \contentsname.).

    bookmark/text=the bookmark text for the table of contents (if empty, no bookmark is set)
    bookmark/bold|bookmark/italic|bookmark/color=the bookmark style for the table of contents
    bookmark/level=the bookmark level: either a number or section or chapter
    Default is chapter, or section if the class does not provide chapters.
}
```

`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).

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

~~~~~ 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` : `\pagesetup`

|  |  |   |
|--|--|---|
| <code>\pagesetup[&lt;pagestyle-name&gt;]{</code>   |  |   |
| <code>norule</code><br><code>rule rule/width[*+ -]=...</code><br><code>rule/color=...</code><br><code>headrule=...</code><br><code>head/rule/width=...</code><br><code>head/rule/color=...</code><br><code>head/rule/font=...</code> | <code>rule/font=...</code><br><code>footrule=...</code><br><code>foot/rule/width=...</code><br><code>foot/rule/color=...</code><br><code>foot/rule/font=...</code> | <code>set \headrulewidth and \footrulewidth to 0pt</code><br><code>set \headrulewidth and \footrulewidth to a given value</code><br><code>To change the rule font/color</code><br><code>redefines \headrule and \footrule (no value restores default)</code><br><code>dimensions</code><br><code>for head and foot rules separately</code><br><code>idem</code> |
| <hr style="border: 0.5px dashed black;"/>  |  |   |
| <code>head=...</code><br><code>left=...</code><br><code>head/left=...</code><br><code>foot/left=...</code><br><code>head/left/even=...</code><br><i>etc.</i>   | <code>foot=...</code><br><code>center=...</code><br><code>head/center=...</code><br><code>foot/center=...</code><br><code>head/left/odd=...</code><br><i>etc.</i>  | <code>right=...</code> , by position<br><code>head/right=...</code> , for one side documents<br><code>foot/right=...</code> , <i>idem</i><br><code>for two side documents</code><br><br><code>changes \fancyhfoffset</code><br><br><code>for two side documents</code><br><br><code>for two side documents</code>   |
| <hr style="border: 0.5px dashed black;"/>  |  |   |
| <code>offset=...</code><br><code>head/offset=...</code><br><code>left/offset=...</code><br><code>head/left/even/offset=...</code><br><i>etc.</i>   | <code>foot/offset=...</code><br><code>right/offset=...</code><br><code>head/left/odd/offset=...</code><br><i>etc.</i>  | <br><br><br><br><br>  |
| <code>foot/right/even/offset=...</code> , <code>foot/right/odd/offset=...</code> ,<br><code>pagestyle=...</code><br><code>\pagesetup* [&lt;pagestyle-name&gt;]{key = value}</code>   |  |   |
| }  |  |   |

`\pagesetup` resets all headers and footers, rules *etc.* and modifies the current (active) page style depending on the `key=value` pairs. `\pagesetup[<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`).

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

`\pagesetup* [<page-style>]` defines or redefines `<page-style>`; the normal form starts from scratch, while the starred form `\pagesetup*` 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  $\text{\LaTeX}$ , 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  $\epsilon$ -TeX `\marks` registers.  $\epsilon$ -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), *ie.* 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>` *ie.* 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 T<sub>E</sub>X shipped out the last page.

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

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

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

```
\getthemarks\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, *ie.* 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` ⇔ there is at most one section title on the current page;
- `\topmarks=\botmarks` ⇔ there is no section title on the current page;
- `\firstmarks=\topmarks` ⇔ 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}{lastsubsection toc}}`.

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`,
- `$\pgfmath$` ( $\leadsto$  `\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>}`  $\leftarrow$  number of columns *or* comma separated list of values  
`text` = the text to be inserted (repeated) in each cell  
`font` = 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 `\\` but you may want `\crrc\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,...,p}`.

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

`\R\row` refer to the value of the list for the current row. `\r` is the same (unless you defined a formula named `r`).  
`\R{<n>}` to the value of the list for row `{<n>}`

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 scaled 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 |

| Mother | Father | Children                           |
|--------|--------|------------------------------------|
| 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)$},
  greeni   = \red,
  greenii  = {\$round(255-\red,0)$},
  greeniii = \red,
  greeniv  = \greenii,
  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 `dcolum` package) and `pgfmth`

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

`\pgfmth` 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

```

\def\weightlist{weight in kg, 30, 45, 50, 55, 60, 65, 70, 80, 90, 95, 100, 110, 120}
\def\heightlist{height in cm, 190, 180, 170, 160, 150, 140, 130}
\colorlet{denutrition}{Crimson} \colorlet{skinniness}{yellow} \colorlet{normal}{LawnGreen}
\colorlet{obesity}{Moccasin} \colorlet{serious obesity}{DarkOrange} \colorlet{morbid obesity}{Fuchsia}
\newcommand*\mycolor[1]{\cellcolor{#1}}
\ifdim#1pt<17pt denutrition\else \ifdim#1pt<19pt skinniness\else \ifdim#1pt<26pt normal\else
\ifdim#1pt<30pt obesity\else \ifdim#1pt<35pt serious obesity\else morbid obesity\fi\fi\fi\fi\fi}}

\begin{tabu}{*{\countvalues\heightlist}{|c|}} \hline
\repeatcell \heightlist {%
  rows = \weightlist,
  height = \ifnum\column>1 \c\else 1\fi,
  weight = \ifnum\row>1 \r\else 1\fi,
  BMI = {\$round(\weight/((\height/100)^2),1)$},
  text = \mycolor{\BMI}\BMI,
  text/row1 = \c, font/row1 = \color{NavyBlue}, cr/row1 = \\ \hline,
  text/col1 = \r, font/col1 = \color{DarkViolet},
} \\ \hline
\end{tabu}\quad
\begin{tabu}{|c|}
\repeatcell{denutrition,skinniness,normal,obesity,serious obesity,morbid obesity}{%
  text=\cellcolor{\c}{\c},\transpose,font=\large\bfseries\bottomstrut}
\end{tabu}

```

| height in cm \ weight in kg | 190  | 180  | 170  | 160  | 150  | 140  | 130  |
|-----------------------------|------|------|------|------|------|------|------|
| 30                          | 8.3  | 9.3  | 10.4 | 11.7 | 13.3 | 15.3 | 17.8 |
| 45                          | 12.5 | 13.9 | 15.6 | 17.6 | 20.0 | 23.0 | 26.6 |
| 50                          | 13.9 | 15.4 | 17.3 | 19.5 | 22.2 | 25.5 | 29.6 |
| 55                          | 15.2 | 17.0 | 19.0 | 21.5 | 24.4 | 28.1 | 32.5 |
| 60                          | 16.6 | 18.5 | 20.8 | 23.4 | 26.7 | 30.6 | 35.5 |
| 65                          | 18.0 | 20.1 | 22.5 | 25.4 | 28.9 | 33.2 | 38.5 |
| 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 |

|                        |
|------------------------|
| <b>denutrition</b>     |
| <b>skinniness</b>      |
| <b>normal</b>          |
| <b>obesity</b>         |
| <b>serious obesity</b> |
| <b>morbid obesity</b>  |

| weight in kg \ height in cm | 30   | 45   | 50   | 55   | 60   | 65   | 70   | 80   | 90   | 95   | 100  | 110  | 120  |
|-----------------------------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 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 [ $\langle environment, level \rangle$ ]{
    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* [ $\langle environment, level \rangle$ ]{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  $\mathcal{S}$  `\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 env
\interitem or (intertext)
\item Second item
\end{enumerate}
```

The example give in real world:

i) First item in the enumerate env  
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  $\langle text \rangle$  to the specified  $\langle width \rangle$ , by means of  $\langle TruncateMarker \rangle$  if the the truncation occurs.

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

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

```
truncate: \truncate{⟨5cm⟩}{⟨Hello world!⟩}
truncetext: \truncetext{⟨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  $\text{text}_1, \text{text}_2, \dots, \text{text}_n$  and adjust them for truncation:

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

Another example of a tabular made with `\repeatcell`

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

```
\truncetext[⟨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  $\text{text}_1, \dots, \text{text}_n$  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  $\{\text{text}_1, \dots, \text{text}_n\}$ . In other words,  $\text{text}_1, \dots, \text{text}_n$  are truncated to a width proportional to their natural widths. (And the sum of their widths will not exceed  $\langle total \ width \rangle$ ).

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
\truncetext{\htruncwidth{1}}{first quite long}
\truncetext{\htruncwidth{2}}{second short}
\truncetext{\htruncwidth{3}}{third}
```

→ first quite...second...t... (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 embedded (*ie.* 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</code> <code>/.estore in cs</code> | like <code>/.store in</code> but with <code>\csname...\endcsname</code>                                                                  |
| <code>/.true</code> <code>/.false</code>               | <code>/.is choice</code> shortcut for boolean keys                                                                                       |
| <code>/.on</code> <code>/.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                                                                                    |
| <code>/.store as code</code>                           | also provides the <code>/key/.number</code> key to get the numeric index of the selected choice.                                         |
| <code>/.estore as code</code>                          | store value ( <i>expanded value</i> ) in the key. Created <code>+</code> and <code>*</code> keys to append 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 ( <code>-</code> <code>+</code> and <code>*</code> 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 ( <code>+</code> <code>-</code> and <code>*</code> 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 <code>+</code> <code>-</code> and <code>*</code> 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 `\providecommand`): 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:

```
color=\color{yellow}      color=\normalcolor
color=blue                 color=[rgb]{1,0,0}
```

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

```
color=yellow              color=default defined by interfaces if not already
color=blue                 color=[rgb]{1,0,0} defined \AtBeginDocument
```

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

```
color=\normalcolor        color=\color{yellow}
color=\color{blue}         color=\color[rgb]{1,0,0}
```

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 `\changeFont` (in `interfaces-LaTeX`). `size` defines `\f@size`, the macro used by `LATEX` 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 `\providecommand`**

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

`\@whiles \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<sup>A</sup>T<sub>E</sub>X

|   | Patched command                                                                                                            | Type | Importance                          | Module                        | ref. page |
|---|----------------------------------------------------------------------------------------------------------------------------|------|-------------------------------------|-------------------------------|-----------|
| ① | <code>\@pkgextension</code><br><code>\@clsextension</code><br><code>\ifpackageloaded</code><br><code>\ifclassloaded</code> |      | minor<br>optimisation or correction | <code>interfaces-LaTeX</code> | 73        |

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

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

This is because L<sup>A</sup>T<sub>E</sub>X defines those commands `\onlypreamble`, while the equivalent macros in `ltxcmds` can be used inside document as well.

|   | Patched command               | Type           | Importance                      | Module                                                             | ref. page |
|---|-------------------------------|----------------|---------------------------------|--------------------------------------------------------------------|-----------|
| ② | <code>\addcontentsline</code> | <i>prepend</i> | important<br>but no side effect | <code>interface-titlesec</code><br><code>interface-bookmark</code> | 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 |
|---|----------------------|------|---------------------------------|--------------------------------|-----------|
| ③ | <code>\@input</code> |      | important<br>but no side effect | <code>interface-scrfile</code> | 73        |

The `\input` primitive (same as L<sup>A</sup>T<sub>E</sub>X 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 |
|---|------------------------|--------------|------------|---------------------------------|-----------|
| ④ | <code>\bookmark</code> | <i>modif</i> | important  | <code>interface-bookmark</code> | 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 (*ie.* `\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 | <a href="#">interface-makecell</a> | 68        |

`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 | <a href="#">interfaces-hypbmsec</a><br><a href="#">interfaces-titlesec</a> | 44        |

In order to get `\lastsectiontitle`, `\lastsectionbookmark` and `\lastsectiontoc` updated at the right place (*ie. 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 | <a href="#">interfaces-enumitem</a> | 68        |

`\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  | <a href="#">interfaces-pgfkeys</a> | 75        |

`\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   [2010/11/21 v2.3 - 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{LaTeX}{\AtEndOfPackage{\LoadInterface{LaTeX}}}
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{hypermsec}
19 \itf@tp{environ}
20 \itf@tp{truncate}
21 \itf@tp{appendix}
22 \itf@tp{embedfile}
23 \if\relax\@optionlist{interfaces.sty}\relax % no option given
24   \ExecuteOptions\@declaredoptions      % all interfaces are active
25 \fi
26 \ProcessOptions
27 </package>
```

### 21.2 † Interfaces-base

```
28 <*base>
29 \ProvidesPackage{interfaces-base}
30   [2010/11/21 v2.3 Basic macros for interfaces package (FC)]
31 \RequirePackage{etex,ltxcmds,etoolbox,pgfkeys,scrfile}
32 \AtEndOfPackage{%
33   \RequirePackage{interfaces-etoolbox}%
34   \RequirePackage{interfaces-scrfile}%
35   \RequirePackage{interfaces-pgfkeys}
36   \RequirePackage{interfaces-tikz}%
37   \RequirePackage{interfaces-umrand}% There is a \BeforePackage here !
38}% \AtEndOfPackage
```

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

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

```

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

56 \newrobustcmd*\itf@Check@Package[1]{\ltx@ifpackageloaded{#1}
57 {}
58 {\PackageError{interfaces-#1}
59   {The package ‘#1’ must be loaded before ‘interfaces-#1’
60   \MessageBreak Please refer to ‘interfaces.pdf’ document
61   \MessageBreak for correct usage of the ‘interfaces’ package}\@ehd
62 }%
63}% \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<sup>A</sup>T<sub>E</sub>X’s `\ifpackageloaded` – this macros is not restricted to the document preamble.

```

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

```

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

```

## 21.2.2 Interfaces patches

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

```

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

```

`\itf@preapptocmd` The need for this macro has risen 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.

```

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

```

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

```

### 21.2.3 Helper macros

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

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

```
131 %% H E L P E R M A C R O S
132 \begingroup\makeatother
133 \csgdef\itf@strip@at#1@{}
134 \endgroup
135 \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 `\noexpandr` 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{}` 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`.

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

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

```
145 %% G E T A N U M B E R (if possible)
146 %% USAGE: \itf@number\count<text>\relax\@nnil (like \@defaultunits)
147 \def\itf@number#1#2\relax\@nnil{% \relax is to mimick the syntax of \@defaultunits
148   \afterassignment\itf@number@#1\number0#2\relax\itf@number@
149 }
150 \def\itf@number@#1\relax#2\itf@number@{%
151   \ifblank{#1}
152     {\itf@number@@{}}#2\itf@number@@\@nnil
153     \csname ltx@ifx\@let@token\itf@number@@
154       first\else second\fi oftwo\endcsname}
155   \ltx@secondoftwo
156 }% \itf@number@
157 \def\itf@number@@#1{\futurelet\@let@token\itf@number@@@}
158 \def\itf@number@@@{%
159   \csname \ifx\@let@token\relax itf@number@@
160     \else remove@to@nnil%
161   \fi \endcsname
162 }% \itf@number@@@
```

## 21.2.5 Units and math units

`\itf@strip@mu` This is the equivalent of L<sup>A</sup>T<sub>E</sub>X's `\strip@pt` but for math units:

```
163%%  U N I T S   F O R   D I M E N S
164 \def\itf@strip@mu{\expandafter\itf@rem@mu\the}
165 {%
166 \@makeother\M\@makeother\U
167 \lowercase{%
168   \gdef\itf@rem@mu#1.#2MU{#1\ifnum#2>\z@.#2\fi}%
169 }}}
```

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

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

```
170%% \skip = \itf@units@mutopt \muskip
171 \def\itf@units@pttomu#1{\gluetomu\glueexpr
172   \glueexpr#1\relax*1800/%
173   \expandafter\itf@strip@dot\the\dimexpr
174     \fontdimen6\textfont2*100\relax.\@nil
175   \relax
176 }% \itf@units@pttomu
177 \def\itf@units@mutopt#1{\glueexpr
178   \mutoglue\muexpr#1\relax*
179   \expandafter\itf@strip@dot\the\dimexpr
180     \fontdimen6\textfont2*100\relax.\@nil/1800
181   \relax
182 }% \itf@units@mutopt
```

`\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.

```
183%% \muskip = \itf@units@mu{ \skip or \muskip or number}
184 {%
185 \@makeother\P\@makeother\T\@makeother\S\@makeother\B
186 \@makeother\C\@makeother\M\@makeother\D\@makeother\I\@makeother\N
187 \lowercase{%
188 \gdef\itf@units@mu#1#2{\begingroup
189   \edef\itf@tp{#2}%
190   \@tfor\x:={PT}{pt}{PC}{pc}{IN}{in}{BP}{bp}%
191     {CM}{cm}{MM}{mm}{DD}{dd}{CC}{cc}{SP}{sp}\do{%
192     \@expandtwoargs\in@\x\itf@tp\ifin@\@break@tfor\fi
193   }%
194   \expandafter\endgroup
195   \ifin@
196     #1=\itf@units@pttomu{#2}%
197   \else
198     \@defaultunits#1 #2mu\relax\@nnil
199   \fi
200 }% \itf@units@mu
201 }}
```

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

```
202%% \skip = \itf@units@pt( \muskip or \skip or number )
203 {%
204 \@makeother\M\@makeother\U
205 \lowercase{%
206 \gdef\itf@units@pt#1#2{\begingroup
207   \@expandtwoargs\in@{MU}{#2}%
208   \ifin@\else\@expandtwoargs\in@{mu}{#2}\fi
209   \expandafter\endgroup
```

```

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

```

216%% glue * 1.5 = glue * 98302 / 65536
217%% glue * 3/2 = glue * 3/2
218\def\itf@units@scale#1!\itf@units@scale@normalize#1/\@nnil/\@nil}
219\def\itf@units@scale@normalize#1/#2/#3\@nil{%
220 \ifx#2\@nnil
221 \number\dimexpr#1pt\relax/65536
222 \else#1/#2
223 \fi
224 }% \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`).

```

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

236 </base>

```

## 21.3 † Interface L<sup>A</sup>T<sub>E</sub>X

```

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

```

242 \newrobustcmd*\changeFont[1]{\itf@pgfkeys
243   \pgfkeys{/itf/LaTeX/changeFont}{#1}\selectfont
244 }% \changeFont
245 \AfterPackage*{hyperref}
246   {\pdfstringdefDisableCommands{\let\changeFont\@gobble}}

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

279 </LaTeX>

```

## 21.4 Interface TikZ ()

```

280 (*tikz)
281 \ProvidesPackage{interfaces-tikz}
282   [2010/11/21 v2.3 load tikz libraries easily - pgf 2.0 & 2.1 (FC)]
283 \RequirePackage{interfaces-base}
284 \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=\@makeoother...}` solves the problem.

```

285%% Some catcode checking
286 \AfterPackage*{tikz}{%
287   \AtEndPreamble{%
288     \tikzset{every picture/.add style={%
289       /utils/exec={\@makeoother!\@makeoother\::}}}%
290   }%
291 }%

```

## 21.4.2 List of pgf/TikZ packages and libraries

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

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

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

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

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

```

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

```

### 21.4.3 pgfkeys definitions for `\usetikz`

```

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

```

`\usetikz`

```

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

```

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

```

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

409 </tikz>

```

## 21.5 Interface `titlesec`: `\sectionformat`

```

410 (*titlesec)
411 \ProvidesPackage{interfaces-titlesec}
412   [2010/11/15 v2.1 pgfkeys macros for titlesec (FC)]
413 \RequirePackage{interfaces-base}
414 \RequirePackage{auxhook}% <fix for \ttl@append (odd/even pages)>
415 \itf@Check@Package{titlesec}
416 \itf@pgfkeys

```

### 21.5.1 `titlesec` modifications

```

417%%  t i t l e s e c  "p a t c h e s"
418\robustify\titleclass \robustify\titleformat \robustify\titlespacing
419\AddLineBeginMainAux{%
420  \string\providecommand\string\ttl@append[2]{}%
421 }
422\BeforeFile*{ttlkeys.def}{%
423  \catcode32 9\relax % ignore spaces
424  \count@\endlinechar\endlinechar\m@ne % and line feeds
425 }
426\AfterFile*{ttlkeys.def}{% restore spaces and line feeds
427  \catcode32 10\relax
428  \endlinechar\count@
429 }

```

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

```

430%%-----
431%% runinifempty => like hang but no paragraph break (like runin) if title is empty
432\def\ttl@runinifempty{%
433  \csname ttl@{ifx\ttl@savetitle\@empty runin\else hang\fi\endcsname
434 }% \ttl@runinifempty

```

### 21.5.2 `\sectionformat`

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

```

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

```

```

444   {\pgfqkeys{/itf/titlesec/#1#2}
445     {/itf/titlesec/@setup={#1}{#2},#4,/itf/titlesec/@update={#1}{#2}}}
446   {\pgfqkeys{/itf/titlesec/#1#2}
447     {/itf/titlesec/@setup={#1}{#2},#4,shape={#3},/itf/titlesec/@update={#1}{#2}}}%
448 }% \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`

```

449 %% private keys @setup and @update
450 \pgfqkeys{/itf/titlesec}{%
451   @setup/.code 2 args=\itf@ttl@setup{#1}{#2},%
452   @update/.code 2 args=\expandafter\itf@ttl@update
453     \csname#1\endcsname{/itf/titlesec/#1}{#2}%
454 }
455 \newrobustcmd*\itf@ttl@setup[2]{%
456   \pgfkeysifdefined{/itf/titlesec/#1#2/shape}{%
457     {\itf@ttl@newsection{#1}{#2}}%
458     \ifcsundef{ttls@#1#2}{}% no previous spacing settings
459     {\expandafter\expandafter\expandafter%
460       \itf@ttl@getspacing\csname ttls@#1#2\endcsname{/itf/titlesec/#1#2}}%
461     \ifcsundef{ttl@#1#2}{}% no previous format settings
462     {\expandafter\expandafter\expandafter%
463       \itf@ttl@getformat\csname ttl@#1#2\endcsname{/itf/titlesec/#1#2}}%
464 }% \itf@ttl@setup
465 \def\itf@ttl@setkey#1#2{\begingroup
466   \def\x{\pgfkeysvalueof{#1}}\def\y{#2}%
467   \expandafter\endgroup\ifx\x\y\else
468     \pgfkeyssetvalue{#1}{#2}\fi
469 }% \itf@ttl@setkey
470 \def\itf@ttl@getspacing#1#2#3#4#5#6{%
471   \itf@ttl@setkey{#6/left}{#1}%
472   \itf@ttl@setkey{#6/right}{#2}%
473   \itf@ttl@setkey{#6/top}{#3}%
474   \itf@ttl@setkey{#6/bottom}{#4}%
475   \itf@ttl@setkey{#6/indent first}{#5}%
476 }% \itf@ttl@getspacing
477 \def\itf@ttl@getformat#1#2{%
478   \csname ltx@\ifx\pgfkeysvalueof#1first\else second\fi oftwo\endcsname
479     {\itf@ttl@get@format{}}
480     {\itf@ttl@get@format{#1}{#2}}%
481 }% \itf@ttl@getformat
482 \def\itf@ttl@get@format#1#2#3#4#5#6#7{%
483   \if\relax\detokenize{#1}\relax\else
484     \pgfkeyssetvalue{#7/shape}{\expandafter\itf@strip@at\string#1}%
485     \fi
486     \itf@ttl@setkey{#7/font}{#2}%
487     \itf@ttl@setkey{#7/label}{#3}%
488     \itf@ttl@setkey{#7/sep}{#4}%
489     \itf@ttl@setkey{#7/before}{#5}%
490     \itf@ttl@setkey{#7/after}{#6}%
491 }% \itf@ttl@get@format

```

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

```

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

```

```

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

```

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

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

```

524%%-----
525%%  t i t l e c l a s s
526\newrobustcmd*\itf@ttl@titleclass{\@ifstar
527  {\itf@ttl@titleclass@i*}
528  {\itf@ttl@titleclass@i}}%
529}% \itf@ttl@titleclass
530\def\itf@ttl@titleclass@i#1#2#3[#4]#5\@nil{\ifblank{#4}
531  {%
532    \expandafter\titleclass\csname#2\endcsname{#3}%
533  }
534  {%
535    \expandafter\titleclass\csname#2\endcsname{#3}[#4]% #4 = sup-level
536    \if\relax#1\relax % not starred
537      \expandafter\expandafter\expandafter%
538      \itf@ttl@titleclass@ii\expandafter\@gobble\string#4\relax{#2}% \@addtoreset...
539    \fi
540  }%
541  \expandafter\providecommand\csname l@#2\endcsname[2]{}% no entry in toc
542}% \itf@ttl@titleclass@i
543\def\itf@ttl@titleclass@ii#1\relax#2{%
544  \ifcsundef{c@#2}
545    {\newcounter{#2}[#1]}% creates counter for #1
546    {\@addtoreset{#2}{#1}}%
547  \csxdef{the#2}{%
548    \expandafter\noexpand\csname the#1\endcsname
549    .\noexpand\@arabic\expandafter\noexpand\csname c@#2\endcsname
550  }% \the#2
551  \ifcsundef{l@#1}{}%
552  \expandafter\providecommand\csname l@#2\endcsname[2]{%
553    \csname l@#1\endcsname{##1}{##2}%
554  }% \l@#2 = \l@newsection is (by default) the same as \l@sup-level
555 }%

```

```
556}% \itf@ttl@titleclass@ii
```

`\itf@ttl@createsection` If `\sectionformat` is invoked with an unknown sectionning command (*ie.* 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.

```
557%%-----
558%%   C r e a t i n g   n e w   s e c t i o n s
559\def\itf@ttl@createsection#1{% sectionning command undefined: temporarily defined
560  \expandafter\titleclass\csname#1\endcsname{straight}[*]%      in straight class
561  \itf@AfterPackage{hypbmsec}{%
562    \AfterPreamble{%
563      \csletcs{hbs@#1}{#1}%
564      \csxdef{#1}{\noexpand\hbs@do\expandafter\noexpand\csname hbs@#1\endcsname}%
565    }%
566  }%
567  \itf@AfterPackage{interfaces-tocloft}{%
568    \itf@cft@newsection{#1}{toc}%
569  }%
570  \itf@AfterPackage{interfaces-hypbmsec}{%
571    \expandafter\let\csname last#1title\endcsname\relax
572    \expandafter\let\csname last#1bookmark\endcsname\relax
573    \expandafter\let\csname last#1toc\endcsname\relax
574  }%
575}% \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.

```
576%%-----
577%%   D e f i n i n g   k e y s   f o r   e a c h   s e c t i o n
578\def\itf@ttl@newsection#1#2{% #1=section, #2=/* or empty
579  \ifcsundef{#1}% sectionning command undefined: temporarily defined in straight class
580    {\itf@ttl@createsection{#1}}
581    {}%
582  \ifblank{#2}{%
583    \pgfqkeys{/itf/titlesec/#1}{% these keys do not exist for starred sections
584      class/.code=\itf@ttl@titleclass{#1}##1[]\@nil,% with \@addtoreset
585      class/.value required,%                          if sup-level is given
586      class*/.code=\itf@ttl@titleclass*{#1}##1[]\@nil,% without \@addtoreset
587      class*/.value required,%                          even if sup-level is given
588      shape/.initial=,% initial shapes set by titlesec.sty for not numbered sections.
589      the#1/.store in cs=the#1,%
590      the#1/.default=\pgfkeysvalueof{/itf/titlesec/#1/label},%
591    }}
592    {% starred section: unless titlesec, we do not put the default shape to hang,
593    \pgfkeysifdefined{/itf/titlesec/#1/shape}%but to the shape of the not starred
594      {\pgfkeysgetvalue{/itf/titlesec/#1/shape}\itf@tp}% sectionning command
595      {\def\itf@tp{hang}}}%
596    \pgfkeyslet{/itf/titlesec/#1#2/shape}\itf@tp%
597  }%
598  \pgfqkeys{/itf/titlesec/#1#2}{%
599    shape/.value required,%
600    shape@cs/.initial=
601    \ifdefined\itf@last@from@next\itf@last@from@next{#1}\fi % interfaces-hypbmsec
602    \csname ttlh@\pgfkeysvalueof{/itf/titlesec/#1#2/shape}\endcsname,% constant
```

```

603         shape@cs/.is private=interface-titlesec,% constant
604     font/.store as code,%
605         format/.alias*=font,%
606     label/.store as code,%
607     sep/.estore as glue,%
608         labelsep/.alias*=sep,%
609     before/.store as code,%
610     after/.store as code,%
611     indent first/.true=\pgfkeyssetvalue{/itf/titlesec/#1#2/indent first}{\@ne},%
612     indent first/.false=\pgfkeyssetvalue{/itf/titlesec/#1#2/indent first}{\z@},%
613         identfirst/.alias=indent first,%
614         indent first=true,%
615     left/.estore as glue,%
616     top/.estore as glue,%
617     bottom/.estore as glue,%
618     right/.estore as glue,%
619     break/.code=\ifblank{##1}
620         {\cslet{#1break}\@empty}
621         {\protected\csdef{#1break}{##1}},%
622     break*/.code=\protected\apptoc{#1break}{##1},%
623     break+/.alias=break*,%
624     mark/.code=\ifblank{##1}
625         {\cslet{#1mark}\@gobble}
626         {\protected\long\csdef{#1mark}####1{##1}},%
627     mark/.default=,%
628 }% \pgfkeys
629 \ifblank{#2}{% no default marks for starred (numberless) \section*
630 \itf@AfterPackage{interfaces-marks}{% requires interfaces-marks
631     \pgfkeys{/itf/titlesec/#1/mark/.default=\markthe{#1}{##1}}}%
632 }% AfterPackage interfaces-mark
633 \itf@AfterPackage{interfaces-bookmark}{% requires interfaces-bookmark
634 \pgfkeys{/itf/titlesec/#1}{% does not exist for starred sections
635     bookmark/.is multiple choice,%
636     bookmark/level/.estore in cs=toclevel@#1,% depth of the bookmark
637     bookmark/level/.default=\ifcsvoid{tll@#1}{99}{\csuse{tll@#1}},%
638     bookmark/level/.blank=\ifcsvoid{toclevel@#1}
639     {\ifcsvoid{tll@#1}{99}{\csuse{tll@#1}}}{\csuse{toclevel@#1}},%
640     bookmark/toclevel/.alias=bookmark/level,%
641     bookmark/bold/.code=\ifblank{##1}
642     {\pgfkeyslet{/itf/titlesec/#1/bookmark/bold}\@empty}
643     {\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/bold}{bold=##1}},%
644     bookmark/bold/.default=true,%
645     bookmark/bold*/.code=\ifblank{##1}
646     {\pgfkeyslet{/itf/titlesec/#1/bookmark/bold*}\@empty}
647     {\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/bold*}{bold*=##1}},%
648     bookmark/bold*/.default=true,%
649     bookmark/italic/.code=\ifblank{##1}
650     {\pgfkeyslet{/itf/titlesec/#1/bookmark/italic}\@empty}
651     {\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/italic}{italic=##1}},%
652     bookmark/italic/.default=true,%
653     bookmark/italic*/.code=\ifblank{##1}
654     {\pgfkeyslet{/itf/titlesec/#1/bookmark/italic*}\@empty}
655     {\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/italic*}{italic*=##1}},%
656     bookmark/italic*/.default=true,%
657     bookmark/color/.code=\ifblank{##1}
658     {\pgfkeyslet{/itf/titlesec/#1/bookmark/color}\@empty}
659     {\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/color}{color=##1}},%
660     bookmark/color/.default=,%
661     bookmark/color*/.code=\ifblank{##1}
662     {\pgfkeyslet{/itf/titlesec/#1/bookmark/color*}\@empty}
663     {\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/color*}{color*=##1}},%
664     bookmark/color*/.default=,%
665     bookmark/on/.code=\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/on}{disable=false},%
666     bookmark/on/.value forbidden,%
667     bookmark/off/.code=\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/on}{disable=true},%

```

```

668         bookmark/off/.value forbidden,%
669         bookmark/none/.alias=bookmark/off,%
670         bookmark/style/.code=\ifblank{##1}
671             {\pgfkeyslet{/itf/titlesec/#1/bookmark/style}\@empty}
672             {\pgfkeyssetvalue{/itf/titlesec/#1/bookmark/style}{style=##1}},%
673         bookmark/style/.default=,%
674         bookmark/.unknown/.code=\pgfkeysEsetvalue{/itf/titlesec/#1/bookmark/other}
675                                     {\pgfkeyscurrentname={##1}},%
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 sectioning command.

```

688 %% bookmarks styles for sectioning 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             {% \bookmarksetup{color*=} overwrites \sectionformat...{bookmark/color*=}
722             \pgfkeysgetvalue{/itf/bookmark/*}\itf@tp
723             \expandafter\bookmarksetup\expandafter{\itf@tp}%
724             }
725             {}%
726     \fi}

```

```

727     {}%
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@leavevmode
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 \pgfkeys{/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=\pgfkeys{/itf/bookmark}{@save={bold=true}},%
792   bold*/.false=\pgfkeys{/itf/bookmark}{@save={bold=false}},%
793   bold*/@blank/.code=\pgfkeys{/itf/bookmark/@restore}{bold},%
794   bold*/.blank=@blank,%
795   italic*/.true=\pgfkeys{/itf/bookmark}{@save={italic=true}},%
796   italic*/.false=\pgfkeys{/itf/bookmark}{@save={italic=false}},%
797   italic*/@blank/.code=\pgfkeys{/itf/bookmark/@restore}{italic},%
798   italic*/.blank=@blank,%
799   color*/.code={\ifblank{#1}
800     {\pgfkeys{/itf/bookmark/@restore}{color}}
801     {\pgfkeys{/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         \pgfkeysifdefined{/itf/bookmark/@save/color}
823         {\pgfkeysgetvalue{/itf/bookmark/@save/color}\BKM@color}
824         }%
825         \pgfkeysifdefined{/itf/bookmark/@save/FontStyle}
826         {\pgfkeysgetvalue{/itf/bookmark/@save/FontStyle}\BS@BKM@FontStyle}
827         }%
828         \pgfkeysifdefined{/itf/bookmark/@save/grouplevel}
829         {% shortcut: \bookmarksetup{bold*} called inside main group (level 0)
830         \ifnum\pgfkeysvalueof{/itf/bookmark/@save/grouplevel}=\z@
831             \global\pgfkeyslet{/itf/bookmark/@save/FontStyle}\@undefined
832             \global\pgfkeyslet{/itf/bookmark/@save/color}\@undefined
833             \global\pgfkeyslet{/itf/bookmark/@save/grouplevel}\@undefined
834             \global\pgfkeyslet{/itf/bookmark/*}\@undefined
835             \global\let\BKM@color\BKM@color
836             \global\let\BS@BKM@FontStyle\BS@BKM@FontStyle
837         \else % restore after each group as needed
838             \pgfkeyslet{/itf/bookmark/@save/FontStyle}\@undefined
839             \pgfkeyslet{/itf/bookmark/@save/color}\@undefined
840             \pgfkeyslet{/itf/bookmark/*}\@undefined
841             \ifnum\currentgrouplevel>\z@
842                 \aftergroup\itf@BKM@restore
843             \else
844                 \pgfkeyslet{/itf/bookmark/@save/grouplevel}\@undefined
845             \fi\fi
846         }
847     },%
848 },%
849 @restore/bold/.code={% bold is "hidden" in \BS@BKM@FontStyle
850     \begingroup
851     \edef\itf@tp{\bookmarkget{bold}}%
852     \pgfqkeys{/itf/bookmark/@restore}{FontStyle}%
853     \count@=\bookmarkget{bold}%
854     \edef\itf@tp{\endgroup
855         \ifnum\itf@tp=\count@
856             \else \noexpand\bookmarksetup{%
857                 bold\ifcase\count@ =false\fi}\fi
858     }\itf@tp},%
859 @restore/italic/.code={% italic is "hidden" in \BS@BKM@FontStyle
860     \begingroup
861     \edef\itf@tp{\bookmarkget{italic}}%
862     \pgfqkeys{/itf/bookmark/@restore}{FontStyle}%
863     \count@=\bookmarkget{italic}%
864     \edef\itf@tp{\endgroup
865         \ifnum\itf@tp=\count@
866             \else \noexpand\bookmarksetup{%
867                 italic\ifcase\count@ =false\fi}\fi
868     }\itf@tp},%
869 }
870 \def\itf@BKM@restore{%
871     \pgfkeysifdefined{/itf/bookmark/@save/grouplevel}
872     {\pgfkeysalso{/itf/bookmark/@restore}}
873     }%
874 }% \itf@BKM@restore
875 %% Plug the BKM keys in the interfaces pgfkeys (not efficient but quickly written...)
876 \define@key{BKM}{disable}[true]{\itf@pgfkeys\pgfkeysalso{/itf/bookmark/disable={#1}}}
877 \define@key{BKM}{disable*}[true]{\itf@pgfkeys\pgfkeysalso{/itf/bookmark/disable*={#1}}}
878 \define@key{BKM}{bold*}[true]{\itf@pgfkeys\pgfkeysalso{/itf/bookmark/bold*={#1}}}
879 \define@key{BKM}{italic*}[true]{\itf@pgfkeys\pgfkeysalso{/itf/bookmark/italic*={#1}}}
880 \define@key{BKM}{color*}[]{\itf@pgfkeys\pgfkeysalso{/itf/bookmark/color*={#1}}}
881 \define@key{BKM}{atend}[true]{\itf@pgfkeys\pgfkeysalso{/itf/bookmark/atend={#1}}}%

```

`\itf@BKM@write` 1) check if the bookmark is disable

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

```

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

```

## 21.7.2 ref and pageref keys

`\itf@BKM@pageref`

```

897%% pageref= key for \bookmark (using refcount : \getpagerefnnumber)
898 \def\itf@BKM@pageref[#1]{%
899   \expandafter\itf@BKM@@pageref
900   \csname r@#1\endcsname{#1}%
901}% \itf@BKM@pageref
902 \def\itf@BKM@@pageref#1#2{%
903   \ifx#1\relax
904     \protect\G@refundefinedtrue
905     \@latex@warning{%
906       Hyper reference ‘#2’ on page \thepage \space undefined
907       \MessageBreak for bookmark destination page
908     }%
909   \fi
910   \edef\BKM@page{\getrefbykeydefault{#2}{page}{1}}%
911}% \itf@BKM@@pageref
912 \define@key{BKM}{pageref}{%
913   \Hy@babelnormalise\itf@BKM@pageref[{#1}]}

```

`\itf@BKM@ref`

```

914%% ref= key for \bookmark
915 \def\itf@BKM@ref[#1]{%
916   \expandafter\itf@BKM@@ref
917   \csname r@#1\endcsname{#1}%
918}% \itf@BKM@ref
919 \def\itf@BKM@@ref#1#2{%
920   \ifx#1\relax
921     \protect\G@refundefinedtrue
922     \@latex@warning{%
923       Hyper reference ‘#2’ on page \thepage \space undefined
924       \MessageBreak for bookmark destination
925     }%
926     \def\BKM@dest{??}%
927   \else
928     \edef\BKM@dest{\expandafter\@fourthoffive#1\@empty\@empty}%
929   \fi
930}% \BKM@@ref
931 \define@key{BKM}{ref}{%
932   \Hy@babelnormalise\itf@BKM@ref[{#1}]}

```

```
933 </bookmark>
```

## 21.8 † Interface `hypbmsec`

### 21.8.1 `interfaces-hypbmsec`

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

### 21.8.2 `interfaces-hypbmsec.code.tex`

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

`\hbs@do (patch)`

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

`\hbs@process (patch)`

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

`\itf@hbs@next`

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

```

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

```

\itf@last@from@next

```

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

1013 \let\lastsectionname\relax
1014 \@for\itf@tp:=part,section,subsection,subsubsection,paragraph,subparagraph%
1015 \do{\expandafter\let\csname last\itf@tp title\endcsname\relax
1016 \expandafter\let\csname last\itf@tp toc\endcsname\relax
1017 \expandafter\let\csname last\itf@tp bookmark\endcsname\relax
1018 }%
1019 \AtBeginDocument{\ifundef\chapter{}}
1020 {\let\lastchaptertitle\relax
1021 \let\lastchaptertoc\relax
1022 \let\lastchapterbookmark\relax}%
1023 }%

1024 </hypbmsec.code.tex>

```

**21.9  Interface fancyhdr : `\pagesetup`**

```

1025 \*fancyhdr
1026 \ProvidesPackage{interfaces-fancyhdr}
1027 [2010/11/21 v2.3 pgfkeys macros for fancyhdr (FC)]
1028 \RequirePackage{interfaces-base}
1029 \itf@Check@Package{fancyhdr}
1030 \itf@pgfkeys
1031 \robustify\fancyhf
1032 \robustify\fancyhfoffset

```

**21.9.1 `\pagesetup`**`\pagesetup`

```

1033 \newrobustcmd*\pagesetup{%
1034   \let\fancy@gb1\relax % Just in case...
1035   \@ifstar
1036     {\itf@pagesetup{}}
1037     {\itf@pagesetup{reset,}}%
1038 }% \pagesetup
1039 \def\itf@pagesetup#1{\@testopt
1040   {\itf@pagesetup{#1}}}%
1041 }% \itf@pagesetup
1042 \def\itf@pagesetup#1[#2]#3{\ifblank{#2}
1043   {% modifies the current page style
1044     \itf@pgfkeys\pgfqkeys{/itf/fancyhdr}{#1#3}\relax\ps@@fancy}%
1045   {% defines or modifies a given page style
1046     \ifblank{#1}
1047       \csappto% star form: keep existing value (if not overwritten)
1048       \csdef% no star form: overwrite (+ reset at \pagestyle time)
1049       {ps@#2}{\itf@pgfkeys\pgfqkeys{/itf/fancyhdr}{#1#3}\relax\ps@@fancy}%
1050   }%
1051 }% \itf@pagesetup
1052 \pgfqkeys{/itf/pagesetup}{%
1053   @current/.is private=interfaces-fancyhdr,%
1054   @style/.is private=interfaces-fancyhdr,%
1055 }

```

`\pagestyle`

```

1056 \renewrobustcmd*\pagestyle[1]{\ifcsundef{ps@#1}
1057   {\itf@undefinedpagestyle{#1}}
1058   {\csname ps@#1\endcsname}%
1059 }% \pagestyle

```

`\thispagestyle`

```

1060 \renewrobustcmd*\thispagestyle[1]{\ifcsundef{ps@#1}
1061   {\itf@undefinedpagestyle{#1}}
1062   {\global\@specialpagetrue\gdef\@specialstyle{#1}}}%
1063 }% \thispagestyle
1064 \newcommand*\itf@undefinedpagestyle[1]{%
1065   \@latex@error{Undefined page style: #1
1066   \MessageBreak your command will be ignored}\@eha
1067 }% \itf@undefinedpagestyle

```

## 21.9.2 Keys for `\pagesetup`

```

1068 \pgfkeys{%
1069   /handlers/.fancy@hf/.code=%
1070   \edef\itf@tp{\pgfkeyscurrentpath}%
1071   \pgfkeysalso{\pgfkeyscurrentpath/.store as code=%}
1072   \pgfkeysalso{\pgfkeyscurrentpath/.append code/.expanded=%
1073     \csdef{f@ncy#1}{%
1074       \noexpand\pgfkeysvalueof{\itf@tp/font}
1075       \noexpand\pgfkeysvalueof{\itf@tp}\noexpand\strut}},%
1076 }% handler for fancyhdr interface
1077 \pgfqkeys{/itf/fancyhdr}{% keys for \pagesetup
1078   code/.code={#1},% execution at expansion of \ps@... => \pgfqkeys{...,code={...},...}
1079   pagestyle/.style={code=\pagestyle{#1}},%
1080   page style/.alias=pagestyle,%
1081   load style/.alias=pagestyle,%
1082   head/rule/.store as dimen in cs=headrulewidth,%
1083   head/rule/.default=.4pt,%
1084   head/rule/.blank=\z@,%
1085   head/rulewidth/.alias*=head/rule,%
1086   headrulewidth/.alias*=head/rule,%
1087   head/rule/width/.alias*=head/rule,%
1088   head/rule/font/.store as code,%
1089   head/rule/color/.style={head/rule/font+=\color{#1}},%
1090   headruler/.code=%
1091     \let\headrule\pgfkeyscurrentvalue
1092     \ifx\headrule\@empty\else
1093       \preto\headrule{\bgroup\pgfkeysvalueof{/itf/fancyhdr/head/rule/font}}%
1094       \appto\headrule{\egroup}%
1095     \fi,%
1096   headruler/.default=%
1097     \if@fancyplain\let\headrulewidth\plainheadrulewidth\fi
1098     \hrule\@height\headrulewidth \@width\headwidth
1099     \vskip-\headrulewidth\relax,%
1100   headrule/.alias=headruler,%
1101   headrule,% sets default
1102   foot/rule/.store as dimen in cs=footrulewidth,%
1103   foot/rule/.default=.4pt,%
1104   foot/rule/.blank=\z@,%
1105   foot/rulewidth/.alias*=foot/rule,%
1106   footrulewidth/.alias*=foot/rule,%
1107   foot/rule/width/.alias*=foot/rule,%
1108   foot/rule/font/.store as code,%
1109   foot/rule/color/.style={foot/rule/font+=\color{#1}},%
1110   footruler/.code=%
1111     \let\footrule\pgfkeyscurrentvalue
1112     \ifx\footrule\@empty\else
1113       \preto\footrule{\bgroup\pgfkeysvalueof{/itf/fancyhdr/head/rule/font}}%
1114       \appto\footrule{\egroup}%
1115     \fi,%
1116   footruler/.default=%
1117     \if@fancyplain\let\footrulewidth\plainfootrulewidth\fi
1118     \vskip-2\footrulewidth
1119     \hrule\@height\footrulewidth \@width\headwidth\vskip\footrulewidth,%
1120   footrule/.alias=footruler,%
1121   footrule,% sets default
1122   rule/.style={head/rule={#1},foot/rule={#1}},%
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   rules/.alias*=rule,rulewidth/.alias*=rule,%
1127   rule/width/.alias*=rule,rules/width/.alias*=rule,%
1128   norule/.style={rule=\z@},%
1129   norule/.value forbidden,%

```

```

1130     norules/.alias=norule,%
1131     rule/font/.style={head/rule/font={#1},foot/rule/font={#1}},%
1132     rule/font+/.style={head/rule/font+={#1},foot/rule/font+={#1}},%
1133     rule/font*/.alias=rule/font+,%
1134     rules/font/.alias=rule/font,%
1135     rules/font+/.alias=rule/font+,%
1136     rules/font*/.alias=rule/font+,%
1137     rule/color/.style={head/rule/color={#1},foot/rule/color={#1}},%
1138     rules/color/.alias=rule/color,%
1139     foot/rule/skip/.store as glue in cs=footruleskip,%
1140     foot/rule/skip/.default=.3\normalbaselineskip,%
1141     foot/rule/skip/.blank=z@,%
1142     foot/ruleskip/.alias*=foot/rule/skip,%
1143     footruleskip/.alias*=foot/rule/skip,%
1144     font/.style={head/font={#1},foot/font={#1}},%
1145     font+/.style={head/font+={#1},foot/font+={#1}},%
1146     font*/.alias=font+,%
1147     color/.style={head/color={#1},foot/color={#1},head/rule/color={#1},foot/rule/color={#1}},%
1148     head/.style={head/left={#1},head/center={#1},head/right={#1}},%
1149     head/font/.style={head/left/font={#1},head/center/font={#1},head/right/font={#1}},%
1150     head/font+/.style={head/left/font+={#1},head/center/font+={#1},head/right/font+={#1}},%
1151     head/font*/.alias=head/font+,%
1152     head/color/.style={head/left/color={#1},head/center/color={#1},head/right/color={#1},he
1153     nohead/.style={head=,head/font=,head/rule=,head/rule/font=},
1154     foot/.style={foot/left={#1},foot/center={#1},foot/right={#1}},%
1155     foot/font/.style={foot/left/font={#1},foot/center/font={#1},foot/right/font={#1}},%
1156     foot/font+/.style={foot/left/font+={#1},foot/center/font+={#1},foot/right/font+={#1}},%
1157     foot/font*/.alias=foot/font+,%
1158     foot/color/.style={foot/left/color={#1},foot/center/color={#1},foot/right/color={#1},fo
1159     nofoot/.style={foot=,foot/font=,foot/rule=,foot/rule/font=},
1160     left/.style={head/left={#1},foot/left={#1}},%
1161     left/font/.style={head/left/font={#1},foot/left/font={#1}},%
1162     left/font+/.style={head/left/font+={#1},foot/left/font+={#1}},%
1163     left/font*/.alias=left/font+,%
1164     left/color/.style={head/left/color={#1},foot/left/color={#1}},%
1165     center/.style={head/center={#1},foot/center={#1}},%
1166     center/font/.style={head/center/font={#1},foot/center/font={#1}},%
1167     center/font+/.style={head/center/font+={#1},foot/center/font+={#1}},%
1168     center/font*/.alias=center/font+,%
1169     center/color/.style={head/center/color={#1},foot/center/color={#1}},%
1170     right/.style={head/right={#1},foot/right={#1}},%
1171     right/font/.style={head/right/font={#1},foot/right/font={#1}},%
1172     right/font+/.style={head/right/font+={#1},foot/right/font+={#1}},%
1173     right/font*/.alias=right/font+,%
1174     right/color/.style={head/right/color={#1},foot/right/color={#1}},%
1175     head/left/.style={head/left/even={#1},head/left/odd={#1}},%
1176     head/left/font/.style={head/left/even/font={#1},head/left/odd/font={#1}},%
1177     head/left/font+/.style={head/left/even/font+={#1},head/left/odd/font+={#1}},%
1178     head/left/font*/.alias=head/left/font+,%
1179     head/left/color/.style={head/left/even/color={#1},head/left/odd/color={#1}},%
1180     left/head/.alias=head/left,%
1181     left/head/font/.alias*=head/left/font,%
1182     left/head/color/.alias=head/left/color,%
1183     head/center/.style={head/center/even={#1},head/center/odd={#1}},%
1184     head/center/font/.style={head/center/even/font={#1},head/center/odd/font={#1}},%
1185     head/center/font+/.style={head/center/even/font+={#1},head/center/odd/font+={#1}},%
1186     head/center/font*/.alias=head/center/font+,%
1187     head/center/color/.style={head/center/even/color={#1},head/center/odd/color={#1}},%
1188     center/head/.alias=head/center,%
1189     center/head/font/.alias*=head/center/font,%
1190     center/head/color/.alias=head/center/color,%
1191     head/right/.style={head/right/even={#1},head/right/odd={#1}},%
1192     head/right/font/.style={head/right/even/font={#1},head/right/odd/font={#1}},%
1193     head/right/font+/.style={head/right/even/font+={#1},head/right/odd/font+={#1}},%
1194     head/right/font*/.alias=head/right/font+,%

```

```

1195     head/right/color/.style={head/right/even/color={#1},head/right/odd/color={#1}},%
1196     right/head/.alias=head/right,%
1197     right/head/font/.alias*=head/right/font,%
1198     right/head/color/.alias=head/right/color,%
1199 foot/left/.style={foot/left/even={#1},foot/left/odd={#1}},%
1200     foot/left/font/.style={foot/left/even/font={#1},foot/left/odd/font={#1}},%
1201     foot/left/font+/.style={foot/left/even/font+={#1},foot/left/odd/font+={#1}},%
1202     foot/left/font*/.alias=foot/left/font+,%
1203     foot/left/color/.style={foot/left/even/color={#1},foot/left/odd/color={#1}},%
1204     left/foot/.alias=foot/left,%
1205     left/foot/font/.alias*=foot/left/font,%
1206     left/foot/color/.alias=foot/left/color,%
1207 foot/center/.style={foot/center/even={#1},foot/center/odd={#1}},%
1208     foot/center/font/.style={foot/center/even/font={#1},foot/center/odd/font={#1}},%
1209     foot/center/font+/.style={foot/center/even/font+={#1},foot/center/odd/font+={#1}},%
1210     foot/center/font*/.alias=foot/center/font+,%
1211     foot/center/color/.style={foot/center/even/color={#1},foot/center/odd/color={#1}},%
1212     center/foot/.alias=foot/center,%
1213     center/foot/font/.alias*=foot/center/font,%
1214     center/foot/color/.alias=foot/center/color,%
1215 foot/right/.style={foot/right/even={#1},foot/right/odd={#1}},%
1216     foot/right/font/.style={foot/right/even/font={#1},foot/right/odd/font={#1}},%
1217     foot/right/font+/.style={foot/right/even/font+={#1},foot/right/odd/font+={#1}},%
1218     foot/right/font*/.alias=foot/right/font+,%
1219     foot/right/color/.style={foot/right/even/color={#1},foot/right/odd/color={#1}},%
1220     right/foot/.alias=foot/right,%
1221     right/foot/font/.alias*=foot/right/font,%
1222     right/foot/color/.alias=foot/right/color,%
1223 head/left/odd/.fancy@hf=olh,%
1224 head/left/odd/font/.store as code,%
1225 head/left/odd/color/.style={head/left/odd/font+=\color{#1}},%
1226     head/odd/left/.alias=head/left/odd,%
1227     head/odd/left/font/.alias*=head/left/odd/font,%
1228     head/odd/left/color/.alias=head/left/odd/color,%
1229     odd/head/left/.alias=head/left/odd,%
1230     odd/head/left/font/.alias*=head/left/odd/font,%
1231     odd/head/left/color/.alias=head/left/odd/color,%
1232     odd/left/head/.alias=head/left/odd,%
1233     odd/left/head/font/.alias*=head/left/odd/font,%
1234     odd/left/head/color/.alias=head/left/odd/color,%
1235 head/center/odd/.fancy@hf=och,%
1236 head/center/odd/font/.store as code,%
1237 head/center/odd/color/.style={head/center/odd/font+=\color{#1}},%
1238     head/odd/center/.alias=head/center/odd,%
1239     head/odd/center/font/.alias*=head/center/odd/font,%
1240     head/odd/center/color/.alias=head/center/odd/color,%
1241     odd/head/center/.alias=head/center/odd,%
1242     odd/head/center/font/.alias*=head/center/odd/font,%
1243     odd/head/center/color/.alias=head/center/odd/color,%
1244     odd/center/head/.alias=head/center/odd,%
1245     odd/center/head/font/.alias*=head/center/odd/font,%
1246     odd/center/head/color/.alias=head/center/odd/color,%
1247 head/right/odd/.fancy@hf=orh,%
1248 head/right/odd/font/.store as code,%
1249 head/right/odd/color/.style={head/right/odd/font+=\color{#1}},%
1250     head/odd/right/.alias=head/right/odd,%
1251     head/odd/right/font/.alias*=head/right/odd/font,%
1252     head/odd/right/color/.alias=head/right/odd/color,%
1253     odd/head/right/.alias=head/right/odd,%
1254     odd/head/right/font/.alias*=head/right/odd/font,%
1255     odd/head/right/color/.alias=head/right/odd/color,%
1256     odd/right/head/.alias=head/right/odd,%
1257     odd/right/head/font/.alias*=head/right/odd/font,%
1258     odd/right/head/color/.alias=head/right/odd/color,%
1259 head/left/even/.fancy@hf=elh,%

```

```

1260 head/left/even/font/.store as code,%
1261 head/left/even/color/.style={head/left/even/font+=\color{#1}},%
1262 head/even/left/.alias=head/left/even,%
1263 head/even/left/font/.alias*=head/left/even/font,%
1264 head/even/left/color/.alias=head/left/even/color,%
1265 even/head/left/.alias=head/left/even,%
1266 even/head/left/font/.alias*=head/left/even/font,%
1267 even/head/left/color/.alias=head/left/even/color,%
1268 even/left/head/.alias=head/left/even,%
1269 even/left/head/font/.alias*=head/left/even/font,%
1270 even/left/head/color/.alias=head/left/even/color,%
1271 head/center/even/.fancy@hf=ech,%
1272 head/center/even/font/.store as code,%
1273 head/center/even/color/.style={head/center/even/font+=\color{#1}},%
1274 head/even/center/.alias=head/center/even,%
1275 head/even/center/font/.alias*=head/center/even/font,%
1276 head/even/center/color/.alias=head/center/even/color,%
1277 even/head/center/.alias=head/center/even,%
1278 even/head/center/font/.alias*=head/center/even/font,%
1279 even/head/center/color/.alias=head/center/even/color,%
1280 even/center/head/.alias=head/center/even,%
1281 even/center/head/font/.alias*=head/center/even/font,%
1282 even/center/head/color/.alias=head/center/even/color,%
1283 head/right/even/.fancy@hf=erh,%
1284 head/right/even/font/.store as code,%
1285 head/right/even/color/.style={head/right/even/font+=\color{#1}},%
1286 head/even/right/.alias=head/right/even,%
1287 head/even/right/font/.alias*=head/right/even/font,%
1288 head/even/right/color/.alias=head/right/even/color,%
1289 even/head/right/.alias=head/right/even,%
1290 even/head/right/font/.alias*=head/right/even/font,%
1291 even/head/right/color/.alias=head/right/even/color,%
1292 even/right/head/.alias=head/right/even,%
1293 even/right/head/font/.alias*=head/right/even/font,%
1294 even/right/head/color/.alias=head/right/even/color,%
1295 foot/left/odd/.fancy@hf=olf,%
1296 foot/left/odd/font/.store as code,%
1297 foot/left/odd/color/.style={foot/left/odd/font+=\color{#1}},%
1298 foot/odd/left/.alias=foot/left/odd,%
1299 foot/odd/left/font/.alias*=foot/left/odd/font,%
1300 foot/odd/left/color/.alias=foot/left/odd/color,%
1301 odd/foot/left/.alias=foot/left/odd,%
1302 odd/foot/left/font/.alias*=foot/left/odd/font,%
1303 odd/foot/left/color/.alias=foot/left/odd/color,%
1304 odd/left/foot/.alias=foot/left/odd,%
1305 odd/left/foot/font/.alias*=foot/left/odd/font,%
1306 odd/left/foot/color/.alias=foot/left/odd/color,%
1307 foot/center/odd/.fancy@hf=ocf,%
1308 foot/center/odd/font/.store as code,%
1309 foot/center/odd/color/.style={foot/center/odd/font+=\color{#1}},%
1310 foot/odd/center/.alias=foot/center/odd,%
1311 foot/odd/center/font/.alias*=foot/center/odd/font,%
1312 foot/odd/center/color/.alias=foot/center/odd/color,%
1313 odd/foot/center/.alias=foot/center/odd,%
1314 odd/foot/center/font/.alias*=foot/center/odd/font,%
1315 odd/foot/center/color/.alias=foot/center/odd/color,%
1316 odd/center/foot/.alias=foot/center/odd,%
1317 odd/center/foot/font/.alias*=foot/center/odd/font,%
1318 odd/center/foot/color/.alias=foot/center/odd/color,%
1319 foot/right/odd/.fancy@hf=orf,%
1320 foot/right/odd/font/.store as code,%
1321 foot/right/odd/color/.style={foot/right/odd/font+=\color{#1}},%
1322 foot/odd/right/.alias=foot/right/odd,%
1323 foot/odd/right/font/.alias*=foot/right/odd/font,%
1324 foot/odd/right/color/.alias=foot/right/odd/color,%

```

```

1325         odd/foot/right/.alias=foot/right/odd,%
1326         odd/foot/right/font/.alias*=foot/right/odd/font,%
1327         odd/foot/right/color/.alias=foot/right/odd/color,%
1328         odd/right/foot/.alias=foot/right/odd,%
1329         odd/right/foot/font/.alias*=foot/right/odd/font,%
1330         odd/right/foot/color/.alias=foot/right/odd/color,%
1331     foot/left/even/.fancy@hf=elf,%
1332     foot/left/even/font/.store as code,%
1333     foot/left/even/color/.style={foot/left/even/font+=\color{#1}},%
1334         foot/even/left/.alias=foot/left/even,%
1335         foot/even/left/font/.alias*=foot/left/even/font,%
1336         foot/even/left/color/.alias=foot/left/even/color,%
1337         even/foot/left/.alias=foot/left/even,%
1338         even/foot/left/font/.alias*=foot/left/even/font,%
1339         even/foot/left/color/.alias=foot/left/even/color,%
1340         even/left/foot/.alias=foot/left/even,%
1341         even/left/foot/font/.alias*=foot/left/even/font,%
1342         even/left/foot/color/.alias=foot/left/even/color,%
1343     foot/center/even/.fancy@hf=ecf,%
1344     foot/center/even/font/.store as code,%
1345     foot/center/even/color/.style={foot/center/even/font+=\color{#1}},%
1346         foot/even/center/.alias=foot/center/even,%
1347         foot/even/center/font/.alias*=foot/center/even/font,%
1348         foot/even/center/color/.alias=foot/center/even/color,%
1349         even/foot/center/.alias=foot/center/even,%
1350         even/foot/center/font/.alias*=foot/center/even/font,%
1351         even/foot/center/color/.alias=foot/center/even/color,%
1352         even/center/foot/.alias=foot/center/even,%
1353         even/center/foot/font/.alias*=foot/center/even/font,%
1354         even/center/foot/color/.alias=foot/center/even/color,%
1355     foot/right/even/.fancy@hf=erf,%
1356     foot/right/even/font/.store as code,%
1357     foot/right/even/color/.style={foot/right/even/font+=\color{#1}},%
1358         foot/even/right/.alias=foot/right/even,%
1359         foot/even/right/font/.alias*=foot/right/even/font,%
1360         foot/even/right/color/.alias=foot/right/even/color,%
1361         even/foot/right/.alias=foot/right/even,%
1362         even/foot/right/font/.alias*=foot/right/even/font,%
1363         even/foot/right/color/.alias=foot/right/even/color,%
1364         even/right/foot/.alias=foot/right/even,%
1365         even/right/foot/font/.alias*=foot/right/even/font,%
1366         even/right/foot/color/.alias=foot/right/even/color,%
1367     offset/.style={head/offset={#1},foot/offset={#1}},%
1368         offset+/.style={head/offset+={#1},foot/offset+={#1}},%
1369         offset-/.style={head/offset-={#1},foot/offset-={#1}},%
1370         offset*/.style={head/offset*={#1},foot/offset*={#1}},%
1371     head/offset/.style={head/left/offset={#1},head/right/offset={#1}},%
1372         head/offset+/.style={head/left/offset+={#1},head/right/offset+={#1}},%
1373         head/offset-/.style={head/left/offset-={#1},head/right/offset-={#1}},%
1374         head/offset*/.style={head/left/offset*={#1},head/right/offset*={#1}},%
1375         offset/head/.alias*=head/offset,%
1376     foot/offset/.style={foot/left/offset={#1},foot/right/offset={#1}},%
1377         foot/offset+/.style={foot/left/offset+={#1},foot/right/offset+={#1}},%
1378         foot/offset-/.style={foot/left/offset-={#1},foot/right/offset-={#1}},%
1379         foot/offset*/.style={foot/left/offset*={#1},foot/right/offset*={#1}},%
1380         offset/foot/.alias*=foot/offset,%
1381     odd/offset/.style={left/odd/offset={#1},right/odd/offset={#1}},%
1382         odd/offset+/.style={left/odd/offset+={#1},right/odd/offset+={#1}},%
1383         odd/offset-/.style={left/odd/offset-={#1},right/odd/offset-={#1}},%
1384         offset/odd/.alias*=odd/offset,%
1385     even/offset/.style={left/even/offset={#1},right/even/offset={#1}},%
1386         even/offset+/.style={left/even/offset+={#1},right/even/offset+={#1}},%
1387         even/offset-/.style={left/even/offset-={#1},right/even/offset-={#1}},%
1388         offset/even/.alias*=even/offset,%
1389     left/offset/.style={foot/left/offset={#1},head/left/offset={#1}},%

```

```

1390     left/offset+/.style={foot/left/offset+={#1},head/left/offset+={#1}},%
1391     left/offset-/.style={foot/left/offset-={#1},head/left/offset-={#1}},%
1392     left/offset*/.style={foot/left/offset*={#1},head/left/offset*={#1}},%
1393     offset/left/.alias*=left/offset,%
1394     left/even/offset/.style={foot/left/even/offset={#1},head/left/even/offset={#1}},%
1395     left/even/offset+/.style={foot/left/even/offset+={#1},head/left/even/offset+={#1}},%
1396     left/even/offset-/.style={foot/left/even/offset-={#1},head/left/even/offset-={#1}},%
1397     left/even/offset*/.style={foot/left/even/offset*={#1},head/left/even/offset*={#1}},%
1398     even/left/offset/.alias*=left/even/offset,%
1399     offset/even/left/.alias*=left/even/offset,%
1400     offset/left/even/.alias*=left/even/offset,%
1401     left/odd/offset/.style={foot/left/odd/offset={#1},head/left/odd/offset={#1}},%
1402     left/odd/offset+/.style={foot/left/odd/offset+={#1},head/left/odd/offset+={#1}},%
1403     left/odd/offset-/.style={foot/left/odd/offset-={#1},head/left/odd/offset-={#1}},%
1404     left/odd/offset*/.style={foot/left/odd/offset*={#1},head/left/odd/offset*={#1}},%
1405     odd/left/offset/.alias*=left/odd/offset,%
1406     offset/odd/left/.alias*=left/odd/offset,%
1407     offset/left/odd/.alias*=left/odd/offset,%
1408     right/offset/.style={foot/right/offset={#1},head/right/offset={#1}},%
1409     right/offset+/.style={foot/right/offset+={#1},head/right/offset+={#1}},%
1410     right/offset-/.style={foot/right/offset-={#1},head/right/offset-={#1}},%
1411     right/offset*/.style={foot/right/offset*={#1},head/right/offset*={#1}},%
1412     offset/right/.alias*=right/offset,%
1413     right/even/offset/.style={foot/right/even/offset={#1},head/right/even/offset={#1}},%
1414     right/even/offset+/.style={foot/right/even/offset+={#1},head/right/even/offset+={#1}},%
1415     right/even/offset-/.style={foot/right/even/offset-={#1},head/right/even/offset-={#1}},%
1416     right/even/offset*/.style={foot/right/even/offset*={#1},head/right/even/offset*={#1}},%
1417     even/right/offset/.alias*=right/even/offset,%
1418     offset/even/right/.alias*=right/even/offset,%
1419     offset/right/even/.alias*=right/even/offset,%
1420     right/odd/offset/.style={foot/right/odd/offset={#1},head/right/odd/offset={#1}},%
1421     right/odd/offset+/.style={foot/right/odd/offset+={#1},head/right/odd/offset+={#1}},%
1422     right/odd/offset-/.style={foot/right/odd/offset-={#1},head/right/odd/offset-={#1}},%
1423     right/odd/offset*/.style={foot/right/odd/offset*={#1},head/right/odd/offset*={#1}},%
1424     odd/right/offset/.alias*=right/odd/offset,%
1425     offset/odd/right/.alias*=right/odd/offset,%
1426     offset/right/odd/.alias*=right/odd/offset,%
1427     head/left/offset/.style={head/left/even/offset={#1},head/left/odd/offset={#1}},%
1428     head/left/offset+/.style={head/left/even/offset+={#1},head/left/odd/offset+={#1}},%
1429     head/left/offset-/.style={head/left/even/offset-={#1},head/left/odd/offset-={#1}},%
1430     head/left/offset*/.style={head/left/even/offset*={#1},head/left/odd/offset*={#1}},%
1431     offset/head/left/.alias*=head/left/offset,%
1432     offset/left/head/.alias*=head/left/offset,%
1433     left/head/offset/.alias*=head/left/offset,%
1434     head/right/offset/.style={head/right/even/offset={#1},head/right/odd/offset={#1}},%
1435     head/right/offset+/.style={head/right/even/offset+={#1},head/right/odd/offset+={#1}},%
1436     head/right/offset-/.style={head/right/even/offset-={#1},head/right/odd/offset-={#1}},%
1437     head/right/offset*/.style={head/right/even/offset*={#1},head/right/odd/offset*={#1}},%
1438     offset/head/right/.alias*=head/right/offset,%
1439     offset/right/head/.alias*=head/right/offset,%
1440     right/head/offset/.alias*=head/right/offset,%
1441     foot/left/offset/.style={foot/left/even/offset={#1},foot/left/odd/offset={#1}},%
1442     foot/left/offset+/.style={foot/left/even/offset+={#1},foot/left/odd/offset+={#1}},%
1443     foot/left/offset-/.style={foot/left/even/offset-={#1},foot/left/odd/offset-={#1}},%
1444     foot/left/offset*/.style={foot/left/even/offset*={#1},foot/left/odd/offset*={#1}},%
1445     offset/foot/left/.alias*=foot/left/offset,%
1446     offset/left/foot/.alias*=foot/left/offset,%
1447     left/foot/offset/.alias*=foot/left/offset,%
1448     foot/right/offset/.style={foot/right/even/offset={#1},foot/right/odd/offset={#1}},%
1449     foot/right/offset+/.style={foot/right/even/offset+={#1},foot/right/odd/offset+={#1}},%
1450     foot/right/offset-/.style={foot/right/even/offset-={#1},foot/right/odd/offset-={#1}},%
1451     foot/right/offset*/.style={foot/right/even/offset*={#1},foot/right/odd/offset*={#1}},%
1452     offset/foot/right/.alias*=foot/right/offset,%
1453     offset/right/foot/.alias*=foot/right/offset,%
1454     right/foot/offset/.alias*=foot/right/offset,%

```

```

1455 head/left/even/offset/.store in skip cs=f@ncy0@elh,%
1456 head/left/even/offset/.append code=\fancy@setoffs,%
1457 offset/head/left/even/.alias*=head/left/even/offset,%
1458 offset/head/even/left/.alias*=head/left/even/offset,%
1459 offset/left/head/even/.alias*=head/left/even/offset,%
1460 offset/left/even/head/.alias*=head/left/even/offset,%
1461 offset/even/left/head/.alias*=head/left/even/offset,%
1462 offset/even/head/left/.alias*=head/left/even/offset,%
1463 left/even/head/offset/.alias*=head/left/even/offset,%
1464 left/head/even/offset/.alias*=head/left/even/offset,%
1465 head/left/odd/offset/.store in skip cs=f@ncy0@olh,%
1466 head/left/odd/offset/.append code=\fancy@setoffs,%
1467 offset/head/left/odd/.alias*=head/left/odd/offset,%
1468 offset/head/odd/left/.alias*=head/left/odd/offset,%
1469 offset/left/head/odd/.alias*=head/left/odd/offset,%
1470 offset/left/odd/head/.alias*=head/left/odd/offset,%
1471 offset/odd/left/head/.alias*=head/left/odd/offset,%
1472 offset/odd/head/left/.alias*=head/left/odd/offset,%
1473 left/odd/head/offset/.alias*=head/left/odd/offset,%
1474 left/head/odd/offset/.alias*=head/left/odd/offset,%
1475 head/right/even/offset/.store in skip cs=f@ncy0@erh,%
1476 head/right/even/offset/.append code=\fancy@setoffs,%
1477 offset/head/right/even/.alias*=head/right/even/offset,%
1478 offset/head/even/right/.alias*=head/right/even/offset,%
1479 offset/right/head/even/.alias*=head/right/even/offset,%
1480 offset/right/even/head/.alias*=head/right/even/offset,%
1481 offset/even/right/head/.alias*=head/right/even/offset,%
1482 offset/even/head/right/.alias*=head/right/even/offset,%
1483 right/even/head/offset/.alias*=head/right/even/offset,%
1484 right/head/even/offset/.alias*=head/right/even/offset,%
1485 head/right/odd/offset/.store in skip cs=f@ncy0@orh,%
1486 head/right/odd/offset/.append code=\fancy@setoffs,%
1487 offset/head/right/odd/.alias*=head/right/odd/offset,%
1488 offset/head/odd/right/.alias*=head/right/odd/offset,%
1489 offset/right/head/odd/.alias*=head/right/odd/offset,%
1490 offset/right/odd/head/.alias*=head/right/odd/offset,%
1491 offset/odd/right/head/.alias*=head/right/odd/offset,%
1492 offset/odd/head/right/.alias*=head/right/odd/offset,%
1493 right/odd/head/offset/.alias*=head/right/odd/offset,%
1494 right/head/odd/offset/.alias*=head/right/odd/offset,%
1495 foot/left/even/offset/.store in skip cs=f@ncy0@elf,%
1496 foot/left/even/offset/.append code=\fancy@setoffs,%
1497 offset/foot/left/even/.alias*=foot/left/even/offset,%
1498 offset/foot/even/left/.alias*=foot/left/even/offset,%
1499 offset/left/foot/even/.alias*=foot/left/even/offset,%
1500 offset/left/even/foot/.alias*=foot/left/even/offset,%
1501 offset/even/left/foot/.alias*=foot/left/even/offset,%
1502 offset/even/foot/left/.alias*=foot/left/even/offset,%
1503 left/even/foot/offset/.alias*=foot/left/even/offset,%
1504 left/foot/even/offset/.alias*=foot/left/even/offset,%
1505 foot/left/odd/offset/.store in skip cs=f@ncy0@olf,%
1506 foot/left/odd/offset/.append code=\fancy@setoffs,%
1507 offset/foot/left/odd/.alias*=foot/left/odd/offset,%
1508 offset/foot/odd/left/.alias*=foot/left/odd/offset,%
1509 offset/left/foot/odd/.alias*=foot/left/odd/offset,%
1510 offset/left/odd/foot/.alias*=foot/left/odd/offset,%
1511 offset/odd/left/foot/.alias*=foot/left/odd/offset,%
1512 offset/odd/foot/left/.alias*=foot/left/odd/offset,%
1513 left/odd/foot/offset/.alias*=foot/left/odd/offset,%
1514 left/foot/odd/offset/.alias*=foot/left/odd/offset,%
1515 foot/right/even/offset/.store in skip cs=f@ncy0@erf,%
1516 foot/right/even/offset/.append code=\fancy@setoffs,%
1517 offset/foot/right/even/.alias*=foot/right/even/offset,%
1518 offset/foot/even/right/.alias*=foot/right/even/offset,%
1519 offset/right/foot/even/.alias*=foot/right/even/offset,%

```

```

1520      offset/right/even/foot/.alias*=foot/right/even/offset,%
1521      offset/even/right/foot/.alias*=foot/right/even/offset,%
1522      offset/even/foot/right/.alias*=foot/right/even/offset,%
1523      right/even/foot/offset/.alias*=foot/right/even/offset,%
1524      right/foot/even/offset/.alias*=foot/right/even/offset,%
1525      foot/right/odd/offset/.store in skip cs=f@ncy0@orf,%
1526      foot/right/odd/offset/.append code=\fancy@setoffs,%
1527      offset/foot/right/odd/.alias*=foot/right/odd/offset,%
1528      offset/foot/odd/right/.alias*=foot/right/odd/offset,%
1529      offset/right/foot/odd/.alias*=foot/right/odd/offset,%
1530      offset/right/odd/foot/.alias*=foot/right/odd/offset,%
1531      offset/odd/right/foot/.alias*=foot/right/odd/offset,%
1532      offset/odd/foot/right/.alias*=foot/right/odd/offset,%
1533      right/odd/foot/offset/.alias*=foot/right/odd/offset,%
1534      right/foot/odd/offset/.alias*=foot/right/odd/offset,%
1535      reset/.style={norule,head=,foot=,offset=\z@,foot/rule/skip,rule/font=,font=},%
1536      reset/.value forbidden,%
1537 }% keys for \pagesetup (stored into token \itf@tk)
1538 \AtBeginDocument{%
1539   \ifdim\headwidth<\z@
1540 %% This catches the case that \headwidth hasn't been initialized and the
1541 %% case that the user added something to \headwidth in the expectation that
1542 %% it was initialized to \textwidth. We compensate this now. This loses if
1543 %% the user intended to multiply it by a factor. But that case is more
1544 %% likely done by saying something like \headwidth=1.2\textwidth.
1545 %% The doc says you have to change \headwidth after the first call to
1546 %% \pagestyle{fancy}. This code is just to catch the most common cases were
1547 %% that requirement is violated.
1548   \global\advance\headwidth123456789sp\global\advance\headwidth\textwidth
1549 \fi}
1550 \</fancyhdr>

```

## 21.10 marks

```

1551 <*marks>
1552 \ProvidesPackage{interfaces-marks}
1553   [2010/11/21 v2.3 easier use of e-TeX marks (FC)]
1554 \RequirePackage{interfaces-base}

```

`\marksthe`

```

1555 \newrobustcmd\marksthe[2]{%
1556   \ifcsname marks@#1\endcsname\else
1557     \itf@marks@new{#1}%
1558   \fi
1559   \expandafter\marks\csname marks@#1\endcsname{\unexpanded{#2}}%
1560 }% \marksthe

```

`\marksthecs`

```

1561 \newrobustcmd*\marksthecs[2]{%
1562   \ifcsname marks@#1\endcsname\else
1563     \itf@marks@new{#1}%
1564   \fi
1565   \ifcsundef{#2}
1566     {}
1567     {\expandafter\marks\csname marks@#1\endcsname{%
1568       \unexpanded\expandafter\expandafter\expandafter{%
1569         \csname#2\endcsname}}}
1570 }% \marksthecs

```

`\itf@marks@new`

```

1571 \def\itf@marks@new#1{%

```

```

1572 \expandafter\newmarks\csname marks@#1\endcsname
1573 \PackageInfo{interfaces-marks}
1574 {New marks ‘#1’ defined}
1575 }% \itf@marks@warning

```

\thefirstmarks

\thebotmarks

\thetopmarks

```

1576 \newcommand*\thefirstmarks[1]{%
1577 \ifcsname marks@#1\endcsname
1578 \expandafter\firstmarks\csname marks@#1\endcsname
1579 \fi
1580 }% \thefirstmarks
1581 \newcommand*\thebotmarks[1]{%
1582 \ifcsname marks@#1\endcsname
1583 \expandafter\botmarks\csname marks@#1\endcsname
1584 \fi
1585 }% \thebotmarks
1586 \newcommand*\thetopmarks[1]{%
1587 \ifcsname marks@#1\endcsname
1588 \expandafter\topmarks\csname marks@#1\endcsname
1589 \fi
1590 }% \thetopmarks

```

\getthemarks

```

1591 %% \getthemarks\firstmarks{section}\macro
1592 \newrobustcmd*\getthemarks[3]{%
1593 \ifcsname marks@#2\endcsname
1594 \expandafter\itf@getthemarks\expandafter#1\csname marks@#2\endcsname{#3}%
1595 \else \let#3\@undefined
1596 \fi
1597 }% \getthemarks
1598 \def\itf@getthemarks#1#2#3{%
1599 \expandafter\def\expandafter#3\expandafter{#1#2}%
1600 }% \itf@getthemarks

```

\getthefirstmarks

\getthebotmarks

\getthetopmarks

```

1601 \newrobustcmd*\getthefirstmarks[2]{\getthemarks\firstmarks{#1}{#2}}
1602 \newrobustcmd*\getthebotmarks[2]{\getthemarks\botmarks{#1}{#2}}
1603 \newrobustcmd*\getthetopmarks[2]{\getthemarks\topmarks{#1}{#2}}

```

\ifmarksequal

```

1604 %% \ifmarksequal\firstmarks\botmarks{subsection}{ true }{ false }
1605 \newrobustcmd*\ifmarksequal[3]{%
1606 \begingroup
1607 \getthemarks{#1}{#3}\@tempa
1608 \getthemarks{#2}{#3}\@tempb
1609 \expandafter\endgroup
1610 \csname ltx@\ifdefined\@tempa\ifdefined\@tempb
1611 \ifx\@tempa\@tempb first\else
1612 second\fi\else
1613 second\fi\else
1614 second\fi
1615 oftwo\endcsname
1616 }% \ifmarksequal

```

1617 </marks>

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

```
1618 <*tocloft>
1619 \ProvidesPackage{interfaces-tocloft}
1620 [2010/11/21 v2.3 pgfkeys macros for tocloft (FC)]
1621 \RequirePackage{interfaces-base}
1622 \itf@Check@Package{tocloft}
1623 \itf@pgfkeys
```

```
\tocsetup The user commands.
\lofsetup
\lotsetup
\listofsetup
1624 \newrobustcmd\tocsetup[1]{%
1625   \AfterPreamble{\itf@pgfkeys\pgfqkeys{/itf/tocloft/toc}{#1}}%
1626 }% \tocsetup
1627 \newrobustcmd\lofsetup[1]{%
1628   \AfterPreamble{\itf@pgfkeys\pgfqkeys{/itf/tocloft/lof}{#1}}%
1629 }% \lofsetup
1630 \newrobustcmd\lotsetup[1]{%
1631   \AfterPreamble{\itf@pgfkeys\pgfqkeys{/itf/tocloft/lot}{#1}}%
1632 }% \lotsetup
1633 \newrobustcmd\listofsetup[2]{%
1634   \AfterPreamble{\itf@pgfkeys\pgfqkeys{/itf/tocloft/#1}{#2}}%
1635 }% \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`.

```
1636 \def\itf@cft@newlistof#1{%
1637   \pgfqkeys{/itf/tocloft/#1}{%
1638     pagestyle/.code=\tocloftpagestyle{##1},%
1639     parskip/.store in skip cs=cftparskip,%
1640     parsep/.alias=parskip,%
1641     mark/.store in cs=cftmark#1,%
1642     marks/.alias=mark,%
1643     label/.store as code,%
1644     ref/.store as code,%
1645     ref/.initial=\csname cft#1title\endcsname,% overwritten (toc,lof,lot)
1646     ref/.default=\csname cft#1title\endcsname,% overwritten (toc,lof,lot)
1647     ref/.blank=\pgfkeysnovalue,%
1648     title/.store as code in cs=cft#1title,% overwritten below for toc, lof and lot
1649     title/top/.store in skip cs=cftbefore#1titleskip,%
1650     beforetitleskip/.alias*=title/top,%
1651     title/bottom/.store in skip cs=cftafter#1titleskip,%
1652     aftertitleskip/.alias*=title/bottom,%
1653     title/font/.store as code in cs=cft#1titlefont,%
1654     title/format/.alias*=title/font,%
1655     title/after/.store as code in cs=cftafter#1title,%
1656     title/command/.code=\csdef{@cftmake#1title}####1{##1},%
1657     dotsep/.code=\itf@units@mu\itf@musk{##1}% converts to mu
1658     \edef\cftdotsep{\itf@strip@mu\itf@musk},%
1659     dotsep/.default=4.5mu,%
1660     dotsep/.blank=\cftnodots,%
1661     dot/.store in=\cftdot,%
1662     before/.store as code,%
1663     after/.store as code,%
1664     columns/before@hook/.is private=interfaces-tocloft,%
1665     columns/after@hook/.is private=interfaces-tocloft,%
1666     columns/before@hook/.initial=,%
1667     columns/after@hook/.initial=,%
1668     columns/.code={\ltx@ifpackageloaded{multicol}}%
1669     \pgfkeysEsetvalue\pgfkeyscurrentkey{\number##1}%
1670     \ifnum\number##1>\@ne
```

```

1671 \pgfkeysgetvalue{/itf/tocloft/#1/columns/before@hook}\itf@tp
1672 \ifx\itf@tp\@empty
1673 \pgfkeyssetvalue{/itf/tocloft/#1/columns/before@hook}{%
1674 \beginingroup
1675 \multicolsep\pgfkeysvalueof{/itf/tocloft/#1/multicolsep}\relax
1676 \begin{multicols}{\pgfkeysvalueof{/itf/tocloft/#1/columns}}%
1677 \columnseprule\pgfkeysvalueof{/itf/tocloft/#1/columns/rule}\relax
1678 \columnsep\pgfkeysvalueof{/itf/tocloft/#1/columns/sep}\relax
1679 \pgfkeysgetvalue{/itf/tocloft/#1/columns/rule/color}\columnseprulecolor
1680 \pgfkeysvalueof{/itf/tocloft/#1/multicols/begin}%
1681 }%
1682 \pgfkeyssetvalue{/itf/tocloft/#1/columns/after@hook}{%
1683 \pgfkeysvalueof{/itf/tocloft/#1/multicols/end}%
1684 \end{multicols}
1685 \endgroup
1686 }%
1687 \fi
1688 \else
1689 \pgfkeyslet\pgfkeyscurrentkey\@ne
1690 \pgfkeyslet{/itf/tocloft/#1/columns/before@hook}\@empty
1691 \pgfkeyslet{/itf/tocloft/#1/columns/after@hook}\@empty
1692 \fi}
1693 {\PackageError{interfaces-tocloft}
1694 {Key ‘columns’: package multicol is required}\@eha}%
1695 },%
1696 columns/.default=2,columns/.blank=1,%
1697 multicol/.alias=columns,%
1698 multicols/.alias=columns,%
1699 twocolumns/.true=\pgfkeysalso{columns=2},%
1700 twocolumns/.false=\pgfkeysalso{columns=1},%
1701 columns/rule/.store as dimen,%
1702 columns/rule/.default=\columnseprule,%
1703 columns/rule,% assign default
1704 columnseprule/.alias*=columns/rule,%
1705 columns/rule/width/.alias*=columns/rule,%
1706 columns/sep/.store as dimen,%
1707 columns/sep/.default=\columnsep,%
1708 columns/sep/.blank=10\p@,%
1709 columns/sep,% assign default
1710 columnsep/.alias*=columns/sep,%
1711 columns/rule/color/.store as color command,%
1712 columns/rule/color,% assign default (\normalcolor)
1713 columnseprulecolor/.alias*=columns/rule/color,%
1714 columns/rulecolor/.alias*=columns/rule/color,%
1715 multicolsep/.store as glue,%
1716 multicolsep/.default=0pt plus4pt minus3pt,%
1717 multicolsep,% sets default
1718 multicols/begin/.initial=,multicols/begin/.default=,%
1719 multicols/before/.alias=multicols/begin,%
1720 multicols/end/.initial=,multicols/end/.default=,%
1721 multicols/after/.alias=multicols/end,%
1722 }% \pgfkeys
1723 \itf@AfterPackage{interfaces-bookmark}{%
1724 \pgfkeys{/itf/tocloft/#1}{%
1725 bookmark/.is multiple choice,%
1726 bookmark/bold/.true=\pgfkeysaddvalue{/itf/tocloft/#1/bookmark}{{},bold=true},%
1727 bookmark/bold/.false=\pgfkeysaddvalue{/itf/tocloft/#1/bookmark}{{},bold=false},%
1728 bookmark/italic/.true=\pgfkeysaddvalue{/itf/tocloft/#1/bookmark}{{},italic=true},%
1729 bookmark/italic/.false=\pgfkeysaddvalue{/itf/tocloft/#1/bookmark}{{},italic=false},%
1730 bookmark/color/.code=\pgfkeysaddvalue{/itf/tocloft/#1/bookmark}{{},color={##1}},%
1731 bookmark/level/.code=\pgfkeysEaddvalue{/itf/tocloft/#1/bookmark}{{},level={%
1732 \ifcsname toclevel@##1\endcsname
1733 \csname toclevel@##1\endcsname
1734 \else##1\fi}},%
1735 bookmark/text/.store as code,%

```

```

1736         bookmark/text/.default=\contentsname,%
1737         bookmark/.unknown/.code=\pgfkeysEaddvalue{/itf/tocloft/#1/bookmark}
1738                                     {{{\pgfkeyscurrentname={##1}}},%
1739         bookmark/.initial=,%
1740     }% \pgfkeys
1741     }% AfterPackage interfaces-bookmark
1742     %% while making a format \jobname is expanded to the name of the format...
1743     %% Therefore, the following commands must be expanded \AtEndPreamble
1744     \AtEndPreamble{%
1745         \BeforeFile{\jobname.#1}{%
1746             \pgfkeysvalueof{/itf/tocloft/#1/before}%
1747             \pgfkeysvalueof{/itf/tocloft/#1/columns/before@hook}%
1748         }%
1749         \AfterFile{\jobname.#1}{%
1750             \pgfkeysvalueof{/itf/tocloft/#1/columns/after@hook}%
1751             \pgfkeysvalueof{/itf/tocloft/#1/after}%
1752         }%
1753     }%
1754     \ifcsname #1setup\endcsname\else
1755         \pgfqkeys{/itf/tocloft/#1}{%
1756             pagenumbers/.true=\@cftpnumon{#1},%
1757             pagenumbers/.false=\@cftpnumoff{#1},%
1758             pagenumber/.alias=pagenumbers
1759         }%
1760         \expandafter\pretocmd\csname @cftmake#1title\expandafter\endcsname
1761             \expandafter{\expandafter\itf@tocloft\csname cft#1title\endcsname{#1}{listof#1}}
1762         {\PackageInfo{interfaces-tocloft}
1763             {Command \expandafter\string\csname @cftmake#1title\endcsname
1764                 successfully augmented}}
1765         {\PackageWarning{interfaces-tocloft}
1766             {Command \expandafter\string\csname @cftmake#1title\endcsname
1767                 \space could not be patched
1768                 \MessageBreak Keys label and bookmark won't work for \string\listofsetup}}}%
1769     \fi
1770 }% \itf@cft@newlistof

```

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

```

1771 \itf@cft@newlistof{toc}
1772 \itf@cft@newlistof{lof}
1773 \itf@cft@newlistof{lot}

1774 %% Specific to toc only -----
1775 \pgfqkeys{/itf/tocloft/toc}{%
1776     depth/.store in counter cs=c@tocdepth,%
1777     depth/.append code=
1778         \global\c@tocdepth\c@tocdepth
1779         \addtocontents{toc}{\c@tocdepth\the\c@tocdepth\relax},%
1780     tocdepth/.alias=depth,%
1781     title/.store as code in cs=contentsname,%
1782     contentsname/.alias=title,%
1783     ref/.initial=\contentsname,%
1784     ref/.default=\contentsname,%
1785     pagenumbers/.style={% boolean key
1786         part/pagenumbers={#1},chapter/pagenumbers={#1},section/pagenumbers={#1},%
1787         subsubsection/pagenumbers={#1},subsubsection/pagenumbers={#1},%
1788         paragraph/pagenumbers={#1},subparagraph/pagenumbers={#1}},%
1789     pagenumber/.alias=pagenumbers,%
1790 }%
1791 %% Specific to lof and lot -----
1792 \pgfqkeys{/itf/tocloft/lof}{%
1793     title/.store as code in cs=listfigurename,%
1794     listfigurename/.alias=title,%
1795     ref/.initial=\listfigurename,%
1796     ref/.default=\listfigurename,%

```

```

1797 }%
1798 \pgfqkeys{/itf/tocloft}{%
1799   title/.store as code in cs=listtablename,%
1800   listtablename/.alias=title,%
1801   ref/.initial=\listtablename,%
1802   ref/.default=\listtablename,%
1803 }%
1804 \if@cftsubfigopt
1805 \pgfqkeys{/itf/tocloft}{%
1806   lof/pagenumbers/.style={figure/pagenumbers={#1},subfigure/pagenumbers={#1}},%
1807   lof/pagenummer/.alias=lof/pagenumbers,%
1808   lof/depth/.store in counter cs=c@lofdepth,%
1809   lof/depth/.append code=\global\c@lofdepth\c@lofdepth
1810   \addtocontents{lof}{\c@lofdepth\the\c@lofdepth\relax},%
1811   lot/pagenumbers/.style={figure/pagenumbers={#1},subfigure/pagenumbers={#1}},%
1812   lot/pagenummer/.alias=lot/pagenumbers,%
1813   lot/depth/.store in counter cs=c@lotdepth,%
1814   lot/depth/.append code=\global\c@lotdepth\c@lotdepth
1815   \addtocontents{lot}{\c@lotdepth\the\c@lotdepth\relax},%
1816 }%
1817 \else
1818 \pgfqkeys{/itf/tocloft}{%
1819   lof/pagenumbers/.style={figure/pagenumbers={#1}},%
1820   lof/pagenummer/.alias=lof/pagenumbers,%
1821   lot/pagenumbers/.style={figure/pagenumbers={#1}},%
1822   lot/pagenummer/.alias=lot/pagenumbers,%
1823 }%
1824 \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**.

```

1825 \def\itf@cft@name#1.#2.#3\@nil{cft#1\itf@cft@secname{#2}#3}
1826 \def\itf@cft@secname#1{%
1827   \expandafter\itf@cft@sec@name\csname#1\endcsname{#1}%
1828 }% \itf@cft@secname
1829 \def\itf@cft@sec@name#1#2{%
1830   \ifx#1\chapter chap\else
1831     \ifx#1\section sec\else
1832       \ifx#1\subsection subsec\else
1833         \ifx#1\subsubsection subsubsec\else
1834           \ifx#1\paragraph para\else
1835             \ifx#1\subparagraph subpara\else
1836               \ifx#1\figure fig\else
1837                 \ifx#1\subsubfigure subfig\else
1838                   \ifx#1\table tab\else
1839                     \ifx#1\subtable subtab\else
1840                       #2\fi
1841   \fi\fi\fi\fi\fi\fi\fi\fi\fi
1842 }% \itf@cft@section@name

```

### 21.11.1 Specific handlers for tocloft

```

1843%% tocloft setup for tables sections :
1844\pgfkeys{%
1845  /handlers/.cft@def/.style={\pgfkeyscurrentpath/.store in cs=\itf@cft@name #1\@nil},%
1846  /handlers/.cft@code/.style={\pgfkeyscurrentpath/.store as code in cs=\itf@cft@name #1\@nil},%
1847  /handlers/.cft@length/.style={\pgfkeyscurrentpath/.store in skip cs=\itf@cft@name #1\@nil},%
1848 }

```

`\itf@cft@newsection` This macro creates the keys for each sectioning command.

```

1849\def\itf@cft@newsection#1#2{% #1=part / chap / sec / subsec etc., #2 = toc | lof | lot
1850  \pgfkeys{/itf/tocloft/#2/#1}%
1851  skip/.cft@length=before.#1.skip,%
1852  before/.alias*=skip,%
1853  top/.alias*=skip,%
1854  indent/.cft@length=#1.indent,%
1855  font/.cft@code=#1.font,%
1856  font/.append code={\pgfkeysalso{/itf/tocloft/#2/#1/pagewidth/font={#1}}},%
1857  format/.alias*=font,%
1858  number/width/.cft@length=#1.numwidth,%
1859  numwidth/.alias*=number/width,%
1860  presnum/.cft@code=#1.presnum,%
1861  number/before/.alias*=presnum,%
1862  aftersnum/.cft@code=#1.aftersnum,%
1863  number/after/.alias*=aftersnum,%
1864  aftersnumb/.cft@code=#1.aftersnumb,%
1865  title/before/.alias*=aftersnumb,%
1866  leader/.cft@def=#1.leader,%
1867  leader/.default=\cftdotfill{\csname\itf@cft@name.#1.dotsep\@nil\endcsname},%
1868  leaders/.alias=leader,%
1869  dotsep/.code={\ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
1870    \csdef{\itf@cft@name.#1.dotsep\@nil}{\cftdotsep}%
1871    \else
1872      \itf@units@mu\itf@musk{#1}%
1873      \csedef{\itf@cft@name.#1.dotsep\@nil}{\itf@strip@mu\itf@musk}%
1874    \fi},%
1875  dotsep/.blank=\cftnodots,%
1876  nodots/.code=\csdef{\itf@cft@name.#1.dotsep\@nil}{\cftnodots},%
1877  nodots/.value forbidden,%
1878  pagefont/.cft@code=#1.pagefont,%
1879  page/font/.alias*=pagefont,%
1880  pagewidth/font/.alias*=pagefont,%
1881  afterpnum/.cft@code=#1.afterpnum,%
1882  page/after/.alias*=afterpnum,%
1883  pagewidth/after/.alias*=afterpnum,%
1884  pagewidths/.on=\@cftpnumon{\itf@cft@secname{#1}},%
1885  pagewidths/.off=\@cftpnumoff{\itf@cft@secname{#1}},%
1886  pagewidth/.alias=pagewidths,%
1887 }% \pgfkeys
1888}% \itf@cft@newsection

1889\@for\itf@tp:=part, chap, chapter, sec, section, subsec, subsection,%
1890      subsubsec, subsubsection, para, paragraph, subpara, subparagraph%
1891  \do{\expandafter\itf@cft@newsection\expandafter{\itf@tp}{toc}}%
1892\if@cftsubfigopt % \ifpackagewith{tocloft}{subfig} ??
1893  \@for\itf@tp:=fig, figure, subfig, subfigure%
1894  \do{\expandafter\itf@cft@newsection\expandafter{\itf@tp}{lof}}%
1895  \@for\itf@tp:=tab, table, subtab, subtable%
1896  \do{\expandafter\itf@cft@newsection\expandafter{\itf@tp}{lot}}%
1897\else
1898  \@for\itf@tp:=fig, figure%
1899  \do{\expandafter\itf@cft@newsection\expandafter{\itf@tp}{lof}}%
1900  \@for\itf@tp:=tab, table%
1901  \do{\expandafter\itf@cft@newsection\expandafter{\itf@tp}{lot}}%

```

1902 \fi

```

\@cftmaketoctitle (patch)
\@cftmakelofttitle (patch)
\@cftmakelotttitle (patch)
\newlistentry (patch)
1903 \itf@patch\pretocmd\@cftmaketoctitle
1904   {\itf@tocloft\contentsname{toc}{tableofcontents}}
1905   {\PackageInfo{interfaces-tocloft}
1906     {Command \string\@cftmaketoctitle\space successfully augmented}}
1907   {\PackageWarning{interfaces-tocloft}
1908     {Command \string\@cftmaketoctitle\space could not be patched
1909       \MessageBreak label and bookmark keys for \string\tocsetup\space won't work}}}%
1910 \itf@patch\pretocmd\@cftmakelofttitle
1911   {\itf@tocloft\listfigurename{lof}{listoffigures}}
1912   {\PackageInfo{interfaces-tocloft}
1913     {Command \string\@cftmakelofttitle\space successfully augmented}}
1914   {\PackageWarning{interfaces-tocloft}
1915     {Command \string\@cftmakelofttitle\space could not be patched
1916       \MessageBreak label and bookmark keys for \string\lofsetup\space won't work}}}%
1917 \itf@patch\pretocmd\@cftmakelotttitle
1918   {\itf@tocloft\listtablename{lot}{listoftables}}
1919   {\PackageInfo{interfaces-tocloft}
1920     {Command \string\@cftmakelotttitle\space successfully augmented}}
1921   {\PackageWarning{interfaces-tocloft}
1922     {Command \string\@cftmakelotttitle\space could not be patched
1923       \MessageBreak label and bookmark keys for \string\lotsetup\space won't work}}}%
1924 \itf@patch\apptocmd\newlistentry
1925   {\itf@cft@newlistof{#2}}
1926   {\PackageInfo{interfaces-tocloft}
1927     {Command \string\newlistentry\space successfully augmented}}
1928   {\PackageWarning{interfaces-tocloft}
1929     {Command \string\newlistentry\space could not be patched
1930       \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.

```

1931 \def\itf@tocloft#1#2#3{%
1932   \expandafter\def\expandafter#1\expandafter{%
1933     \expandafter\itf@cft@label
1934     \expandafter#1\expandafter{#1}{#2}{#3}}}%
1935 }% \itf@tocloft
1936 \long\def\itf@cft@label#1#2#3#4{%
1937   \def#1{#2}% restore original
1938   \begingroup
1939   \pgfkeysifdefined{/itf/tocloft/#3/bookmark}{
1940     {%
1941       \raisedhyperdef{interfaces}{#4}{#2}%
1942       \pgfkeysgetvalue{/itf/tocloft/#3/bookmark/text}\itf@tp
1943       \ifx\itf@tp\empty\else
1944         \pgfkeysgetvalue{/itf/tocloft/#3/bookmark}\@tempa
1945         \expandafter\bookmark\expandafter[%
1946           \expandafter{\@tempa,dest=interfaces.#4}}%
1947         {\pgfkeysvalueof{/itf/tocloft/#3/bookmark/text}}}%
1948       \global\BKM@startatroottrue
1949     \fi}
1950   {#2}%
1951   \pgfkeysgetvalue{/itf/tocloft/#3/label}\itf@tp
1952   \ifx\itf@tp\empty\else
1953     \def\@currentHref{interfaces.#4}%
1954     \pgfkeysgetvalue{/itf/tocloft/#3/ref}\@currentlabel
1955     \def\@currentlabelname{#2}%
1956     \label{\pgfkeysvalueof{/itf/tocloft/#3/label}}}%
1957   \fi
1958 \endgroup

```

```

1959 \gdef\itf@tp{\def#1{#2}\let\itf@tp\@undefined}% restore original
1960 \aftergroup\itf@tp % simple group introduced by tocloft to print \contentsname
1961 }% \itf@cft@label

1962 </tocloft>

```

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

```

1963 <*makecell>
1964 \ProvidesPackage{interfaces-makecell}
1965 [2010/11/21 v2.3 repeatcell to fill in tabulars (FC)]
1966 \RequirePackage{interfaces-base}
1967 \itf@Check@Package{makecell}
1968 \edef\itf@AtEnd{\catcode126 \the\catcode126}% 126 = ~
1969 \AtEndOfPackage{\itf@AtEnd}
1970 \catcode126 \active % 126 = ~
1971 \itf@pgfkeys

```

### 21.12.1 `countvalues`

`\countvalues`

```

1972 \def\countvalues#1{\number\numexpr
1973 \expandafter\itf@countvalues\expandafter0#1,\countvalues\@nnil
1974 }% \countvalues
1975 \def\itf@countvalues#1#2,#3\@nnil{%
1976 \ifx\countvalues#3%
1977 +#1\expandafter\remove@to@nnil
1978 \else
1979 +1\expandafter\itf@countvalues
1980 \fi#3\@nnil
1981 }% \itf@countvalues

```

`\itf@getvalue`

```

1982 \def\itf@getvalue#1#2{\romannumeral-'\q
1983 \expandafter\itf@getv@lue\expandafter0#1,\itf@getvalue\@nil{#2-1}\@nnil
1984 }% \itf@getvalue
1985 \def\itf@getv@lue#1#2,#3\@nil#4\@nnil{%
1986 \ifnum\numexpr#1=\numexpr#4\relax
1987 #2\expandafter\remove@to@nnil
1988 \else\ifx\itf@getvalue#3%
1989 \expandafter\expandafter\expandafter\remove@to@nnil
1990 \else
1991 \expandafter\expandafter\expandafter\itf@getv@lue
1992 \fi\fi{#1+1}#3\@nil#4\@nnil
1993 }% \itf@getv@lue

```

### 21.12.2 `\repeatcell`

`\repeatcell`

```

1994 \newif\ifitf@celltranspose
1995 \def\itf@ampactive{%
1996 \def\tab{&}%
1997 \lccode'\~{38}\lowercase{\def~{&}}%
1998 }% \itf@ampactive
1999 \newrobustcmd*\repeatcell{\begingroup
2000 \let\tab\relax
2001 \catcode38\active
2002 \itf@repeatcell
2003 }% \repeatcell

```

```

2004 \newrobustcmd*\itf@repeatcell[3][]{%
2005   \ifblank{#1}% alignment option ?
2006     {\let\itf@makecell\@firstofone}
2007     {\def\itf@makecell##1{\noexpand\makecell[{#1}]{##1}}}%
2008   \countdef\itf@col=100% local
2009   \countdef\itf@row=101% local
2010   \countdef\itf@cnt=102% local
2011   \toksdef\itf@cell=1% local
2012   \itf@pgfkeys
2013   \pgfqkeys{/itf/makecell/repeatcell}{@reset={#2},#3}%
2014   \toks@{}% local
2015   \ifnum\itf@row>\z@ % number
2016     \edef\itf@rowloop{\noexpand\@whilenum\itf@row<\the\itf@row}%
2017     \def\itf@cell@init{\def\r{\row}\let\R@empty}%
2018   \else % list
2019     \def\itf@rowloop{\@for\r:=\itf@rows}%
2020     \expandafter\def\expandafter\itf@cell@init\expandafter{%
2021       \expandafter\def\expandafter\R\expandafter{%
2022         \expandafter\itf@getvalue\expandafter{\expandafter\@empty\itf@rows}}}%
2023   \fi
2024   \ifnum\itf@col>\z@ % number
2025     \edef\itf@colloop{\noexpand\@whilenum\itf@col<\the\itf@col}%
2026     \expandafter\def\expandafter\itf@cell@init\expandafter{%
2027       \itf@cell@init\def\c{\column}\let\C@empty}%
2028   \else % list
2029     \def\itf@colloop{\@for\c:=\itf@cols}%
2030     \expandafter\expandafter\expandafter\def
2031       \expandafter\expandafter\expandafter\itf@cell@init
2032       \expandafter\expandafter\expandafter{%
2033         \expandafter\itf@cell@init
2034         \expandafter\def\expandafter\C\expandafter{%
2035           \expandafter\itf@getvalue\expandafter{\expandafter\@empty\itf@cols}}}%
2036   \fi
2037   \itf@cell@init % in case they are used in formulas
2038   \ifitf@celltranspose
2039     \itf@col\z@
2040     \itf@colloop\do{%
2041       \ifnum\itf@col>\z@
2042         \pgfkeysifdefined{/itf/makecell/repeatcell/cr/row\the\itf@col}%
2043           {\pgfkeysgetvalue{/itf/makecell/repeatcell/cr/row\the\itf@col}\itf@thecr}
2044           {\pgfkeysgetvalue{/itf/makecell/repeatcell/cr}\itf@thecr}%
2045         \toks@\expandafter\expandafter\expandafter{%
2046           \expandafter\the\expandafter\toks@\itf@thecr}%
2047       \fi
2048       \advance\itf@col\@ne
2049       \itf@row\z@
2050       \itf@rowloop\do{%
2051         \advance\itf@row\@ne
2052         \ifnum\itf@row>\@ne
2053           \toks@\expandafter{\the\toks@&}%
2054         \fi
2055         \toks@\expandafter\expandafter\expandafter{%
2056           \expandafter\the\expandafter\toks@
2057           \itf@cell@init\itf@ampactive}%
2058         \itf@cell@insert}%
2059     }%
2060   \else
2061     \itf@row\z@
2062     \itf@rowloop\do{%
2063       \ifnum\itf@row>\z@
2064         \pgfkeysifdefined{/itf/makecell/repeatcell/cr/row\the\itf@row}%
2065           {\pgfkeysgetvalue{/itf/makecell/repeatcell/cr/row\the\itf@row}\itf@thecr}
2066           {\pgfkeysgetvalue{/itf/makecell/repeatcell/cr}\itf@thecr}%
2067         \toks@\expandafter\expandafter\expandafter{%
2068           \expandafter\the\expandafter\toks@\itf@thecr}%

```

```

2069 \fi
2070 \advance\itf@row\@ne
2071 \itf@col\z@
2072 \itf@colloop\do{%
2073 \advance\itf@col\@ne
2074 \ifnum\itf@col>\@ne
2075 \toks@\expandafter{\the\toks@&}%
2076 \fi
2077 \toks@\expandafter\expandafter\expandafter{%
2078 \expandafter\the\expandafter\toks@
2079 \itf@cell@init\itf@ampactive}%
2080 \itf@cell@insert}%
2081 }%
2082 \fi
2083 \itf@message{Code introduced by repeatcell in the \@currenvir:^^J
2084 \the\toks@}% (to see the code...)
2085 \expandafter\endgroup\the\toks@
2086 \ignorespaces
2087 }% \itf@repeatcell

```

`\itf@message`

```
2088 \let\itf@message\@gobble
```

Now the keys for `\repeatcell`:

```

2089 \def\itf@cell@rowcolkey{%
2090 \removefromstart\pgfkeyscurrentname{row}%
2091 \ifin@ \else
2092 \removefromstart\pgfkeyscurrentname{col}%
2093 \fi
2094 \ifin@
2095 \itf@number\itf@cnt\pgfkeyscurrentname\relax\@nnil
2096 {\pgfkeyslet\pgfkeyscurrentkey\pgfkeyscurrentvalue}
2097 \in@false
2098 \fi
2099 \ifin@ \else
2100 \pgfkeysvalueof{/errors/unknown key/.@cmd}\pgfkeyscurrentkey
2101 \pgfkeyscurrentvalue
2102 \pgfeov
2103 \fi
2104 }% \itf@cell@rowcolkey
2105 \pgfkeys{/itf/makecell/repeatcell}{%
2106 font/.store in=\itf@cellfont,%
2107 font/.unknown/.code=\itf@cell@rowcolkey,%
2108 text/.store in=\itf@celltext,%
2109 text/.default=\ifdefined\C\C\column\fi,%
2110 text/.unknown/.code=\itf@cell@rowcolkey,%
2111 cr/.initial=\\,%
2112 cr/.default=\crr,cr/.blank=\\,end/.alias=cr,%
2113 cr/.unknown/.code=\itf@cell@rowcolkey,%
2114 rows/.code=\itf@number\itf@row#1\relax\@nnil
2115 {\ifnum\itf@row<\z@\itf@row\@ne\fi}
2116 {\itf@row\z@
2117 \def\itf@rows{#1}\itf@expandonceifcs\itf@rows},%
2118 rows/.default=1,rows/.blank=1,%
2119 @cols/.is private=interfaces-makecell,%
2120 column/.is private=interfaces-makecell,%
2121 row/.is private=interfaces-makecell,%
2122 transpose/.is if=\itf@celltranspose,%
2123 .unknown/.code=%
2124 \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text \itf@cell@formulaerror{}\z@ \fi
2125 \ifx\pgfkeyscurrentvalue\@empty \itf@cell@formulaerror{}\@ne \fi
2126 \pgfkeysifdefined{\pgfkeyscurrentkey/@calc}%

```

```

2127         {\itf@cell@formulaerror{}}\tw@}
2128         {\pgfkeyslet{\pgfkeyscurrentkey/@calc}\relax}%
2129         \itf@cell@getformula{#1}%
2130         \pgfkeysEaddvalue{/itf/makecell/repeatcell/@calc}{}}
2131         {\the\toks@ \expandafter \noexpand \csname \pgfkeyscurrentname \endcsname}%
2132         \expandafter \let \csname \pgfkeyscurrentname \endcsname \@undefined,%
2133         @reset/.code=% #1 are the columns
2134         \itf@cell@transposefalse
2135         \let\itf@cell@text\@empty
2136         \let\itf@cell@font\@empty
2137         \pgfkeyssetvalue{/itf/makecell/repeatcell/cr}{\}%
2138         \itf@row\@ne
2139         \pgfkeyssetvalue{/itf/makecell/repeatcell/@calc}{%
2140         \itf@cell@numcalc\itf@col\column
2141         \itf@cell@numcalc\itf@row\row}%
2142         \itf@number\itf@col#1\relax\@nnil
2143         {\ifnum\itf@col<z@\itf@col\@ne\fi}
2144         {\itf@col\z@
2145         \def\itf@cols{#1}\itf@expandonceifcs\itf@cols},%
2146 }% keys for \repeatcell

```

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

```

2147 \def\itf@cell@getformula#1{%
2148   \itf@cell@getformula #1$$$ \itf@cell@getformula $\relax\@nnil
2149   \expandafter \ifx \expandafter \toks@ \the \toks@
2150   \itf@cell@formulaerror{#1}\thr@@
2151   \fi
2152 }% \itf@cell@getformula
2153 \def\itf@cell@getformula #1$#2$#3$#4$#5$#6{%
2154   \toks@{\toks@}%
2155   \ifx\itf@cell@getformula #4%
2156     \toks@{\itf@cell@numcalc{#1}}%
2157   \else\ifx $#4%
2158     \ifx\itf@cell@getformula #6%
2159       \ifdefined\pgfmathparse\else\itf@cell@formulaerror{#2}4\fi
2160       \toks@{\itf@cell@pgfcalc{#2}}%
2161     \else\ifx $#6%
2162       \ifdefined\FPeval\else\itf@cell@formulaerror{#3}5\fi
2163       \toks@{\itf@cell@FPcalc{#3}}%
2164     \fi\fi
2165   \fi\fi\remove@to\@nnil
2166 }% \itf@cell@getformula
2167 \def\itf@cell@formulaerror#1#2{%
2168   \def\itf@tp{#1}\@onelevel@sanitize\itf@tp
2169   \ifcase#2% 0
2170     \PackageError{interfaces-makecell}
2171       {Missing formula for key \pgfkeyscurrentname!}\@ehd
2172   \or %1
2173     \PackageError{interfaces-makecell}
2174       {Computed key \pgfkeyscurrentname\space wes left blank!}\@ehd
2175   \or %2
2176     \PackageError{interfaces-makecell}
2177       {Duplicate definition for computed key \pgfkeyscurrentname
2178       \MessageBreak \string\repeatcell\space will redefine the key}\@eha
2179   \or %3
2180     \PackageError{interfaces-makecell}
2181       {Incorrect formula in \string\repeatcell
2182       \MessageBreak Check: \itf@tp
2183       \MessageBreak Take care of commas and equal signs...}\@ehd
2184   \or %4
2185     \PackageError{interfaces-makecell}
2186       {Formula: \itf@tp
2187       \MessageBreak requires \string\pgfmathparse:\space\string\usetikz\pgfmath}}\@ehd
2188   \or %5

```

```
2189 \PackageError{interfaces-makecell}
2190   {Formula: \itf@tp
2191   \MessageBreak requires \string\FPeval:\space\string\usepackage{fp}}\@ehd
2192 \else
2193 \PackageError{interfaces-makecell}
2194   {Impossible value!}\@ehd
2195 \fi
2196}% \itf@cell@formulaerror
```

```

\itf@cell@calc
\itf@cell@numcalc
\itf@cell@pgfcalc
\itf@cell@FPcalc
2197 \protected\def\itf@cell@calc#1#2{%
2198   \edef#1{\def\noexpand#1{#2}}%
2199   \toks@\expandafter\expandafter\expandafter{%
2200     \expandafter\the\expandafter\toks@#1}%
2201   #1%
2202 }% \itf@cell@calc
2203 \protected\def\itf@cell@numcalc#1#2{%
2204   \itf@cell@calc#2{\number\numexpr#1}%
2205 }% \itf@cell@numcalc
2206 \protected\def\itf@cell@pgfcalc#1#2{%
2207   \pgfmathparse{#1}%
2208   \itf@cell@calc#2\pgfmathresult
2209 }% \itf@cell@pgfcalc
2210 \protected\def\itf@cell@FPcalc#1#2{%
2211   \begingroup
2212   \setbox\z@\vbox{%\FPeval introduces some weird code, hence the calculation in a \vbox
2213     \FPeval\FPresult{#1}%
2214     \global\let\FPresult\FPresult}%
2215   \endgroup
2216   \itf@cell@calc#2\FPresult
2217 }% \itf@cell@FPcalc

```

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

```

2218 \def\itf@cell@getifdefined#1#2{%
2219   \pgfkeysifdefined{/itf/makecell/repeatcell/#1}
2220     {\pgfkeysgetvalue{/itf/makecell/repeatcell/#1}{#2}
2221       {\let#2\empty}}%
2222 }%
2223 \def\itf@cell@insert{%
2224   \pgfkeysvalueof{/itf/makecell/repeatcell/@calc}%
2225   \ifx\R\@empty\else
2226     \toks@\expandafter\expandafter\expandafter{%
2227       \expandafter\the\expandafter\toks@
2228       \expandafter\def\expandafter\r\expandafter{\r}}%
2229   \fi
2230   \ifx\C\@empty\else
2231     \toks@\expandafter\expandafter\expandafter{%
2232       \expandafter\the\expandafter\toks@
2233       \expandafter\def\expandafter\c\expandafter{\c}}%
2234   \fi
2235   \itf@cell@getifdefined{font/row\the\itf@row}\itf@cellfontrow
2236   \itf@cell@getifdefined{font/col\the\itf@col}\itf@cellfontcol
2237   \itf@cell@getifdefined{text/row\the\itf@row}\itf@cellrow
2238   \itf@cell@getifdefined{text/col\the\itf@col}\itf@cellcol
2239   \toks@\expandafter\expandafter\expandafter{%
2240     \expandafter\the\expandafter\toks@\itf@cellfont}%
2241   \ifx\itf@cellrow\@empty
2242     \ifx\itf@cellcol\@empty
2243       \itf@cell\expandafter\expandafter\expandafter{%
2244         \expandafter\itf@cellfontcol\itf@cellfontrow}%
2245       \let\itf@tp\itf@celltext
2246     \else
2247       \itf@cell\expandafter{\itf@cellfontcol}%
2248       \let\itf@tp\itf@cellcol
2249     \fi
2250   \else
2251     \ifx\itf@cellcol\@empty
2252       \itf@cell\expandafter{\itf@cellfontrow}%
2253       \let\itf@tp\itf@cellrow
2254     \else
2255       \itf@cell{}%

```

```

2256      \ifitf@celltranspose
2257      \edef\itf@tp{\noexpand\itf@diaghead(-2,1)
2258        {\expandonce\itf@cellfontrow\expandonce\itf@cellrow}
2259        {\expandonce\itf@cellfontcol\expandonce\itf@cellcol}}%
2260      \else
2261      \edef\itf@tp{\noexpand\itf@diaghead(-2,1)
2262        {\expandonce\itf@cellfontcol\expandonce\itf@cellcol}
2263        {\expandonce\itf@cellfontrow\expandonce\itf@cellrow}}%
2264      \fi
2265    \fi
2266  \fi
2267  \itf@cell\expandafter\expandafter\expandafter{%
2268    \expandafter\the\expandafter\itf@cell\itf@tp}%
2269  \edef\itf@tp{\itf@makecell{\the\itf@cell}}%
2270  \toks@\expandafter\expandafter\expandafter{%
2271    \expandafter\the\expandafter\toks@\itf@tp}%
2272 }% \itf@cell@insert

```

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

```

2273 \def\itf@diaghead(#1)#2#3{%
2274   \begingroup
2275   \edef\itf@tp{%
2276     \noexpand\diaghead({#1}){\unexpanded{#2 #3}}%
2277       {\unexpanded{#2}}
2278       {\unexpanded{#3}}}%
2279   \expandafter\endgroup\itf@tp
2280 }% \itf@diaghead

```

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

`\CT@cellc`

```

2281 \AfterPackage*{colortbl}{%
2282 \def\CT@cellc#1[#2]#3{% #1 is either \@firstofone or \@gobble, [#2]#3 = color spec
2283   \xdef\CT@cell@color{\noexpand\CT@color#1[#{#2}]}{#3}%
2284   \global\let\noexpand\CT@cell@color\relax}%
2285 }}
2286 \makecell

```

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

```

2287 (*enumitem)
2288 \ProvidesPackage{interfaces-enumitem}
2289   [2010/11/15 v2.1 more control on lists (FC)]
2290 \RequirePackage{interfaces-base}
2291 \itf@Check@Package{enumitem}
2292 \itf@pgfkeys

```

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

```

2293 \providerobustcmd*\interitem{%
2294   \item[]\hskip-\leftmargin
2295 }% \interitem

```

`\setlist` (extension, star form)

```

2296 \renewrobustcmd*\setlist{\begingroup \@ifstar
2297   {\let\do\csgappto \itf@setlist}
2298   {\let\do\csgdef \itf@setlist}%

```

```

2299}% \setlist
2300\newrobustcmd*{\itf@setlist[2][0]}%
2301  \let\enit@a\@empty
2302  \let\enit@b\@empty
2303  \@for\x:=#1\do{%
2304    \ifx\x\@empty\else
2305      \itf@number\count@\x\relax\@nnil
2306      {\edef\enit@b{\enit@b,\the\count@}}
2307      {\edef\enit@a{\enit@a,\x}}%
2308    \fi}%
2309  \ifx\enit@a\@empty\def\enit@a{list}\fi
2310  \ifx\enit@b\@empty\def\enit@b{0}\fi
2311  \@for\a:=\enit@a\do{%
2312    \ifx\a\@empty\else
2313      \@for\b:=\enit@b\do{%
2314        \ifx\b\@empty\else
2315          \ifcsundef{end\a}
2316            {\expandafter\itf@setlist@error\expandafter{\a}}
2317            {\do{enit@@\a\romannumeral\b}{\,#2}}}%
2318        \fi}%
2319      \fi}
2320  \endgroup
2321}% \itf@setlist
2322\def\itf@setlist@error#1{%
2323  \PackageError{interfaces-enumitem}
2324    {Environment ‘#1’ is undefined
2325    \MessageBreak Use \string\newlist\space before \string\setlist
2326    \MessageBreak Your command will be ignored}\@eha
2327}% \itf@setlist@error

2328</enumitem>

```

## 21.14 † Interface ⇄ truncate

```

2329<*truncate>
2330\ProvidesPackage{interfaces-truncate}

```

`\truncetext` `\truncetext` is like `\truncate` but leaves the text to its natural width if possible.

```

2331\newrobustcmd\truncetext[3][\TruncateMarker]{% do nothing if width of #3 is less than #2
2332  \begingroup
2333    \hbadness\maxdimen\hfuzz\vsiz
2334    \setbox\z@=\hbox to\dimexpr#2\relax{\ignorespaces#3\unskip}%
2335    \ifnum\badness>\@M
2336      \def\next{\truncate[{\#1}]{\#2}{\#3}}%
2337    \else \def\next{\#3}%
2338    \fi
2339    \expandafter\endgroup\next
2340}% \truncetext

```

`\htruncadjust`

```

2341\newrobustcmd\htruncadjust[1]{\@testopt
2342  {\itf@htruncadjust{\#1}}}%
2343}% \htruncadjust
2344\long\def\itf@htruncadjust#1[#2]#3{%#1=total width, #2=[x_1..x_n], #3=text_1..text_n
2345  \begingroup
2346  \let\itf@trunc@width\@empty
2347  \let\itf@istruncated\@gobble % to remove the first comma
2348  \let\the@dimen\relax
2349  \count@ \z@ \dimen@ \z@
2350  \ifblank{\#2}% no optional [x_1,...,x_n] parameters
2351    {%
2352      \@for\x:=#3\do{%

```

```

2353         \advance \count@ \@ne
2354         \setbox\z@=\hbox{\ignorespaces\x\unskip}%
2355         \dimen\count@ = \wd\z@
2356         \toks\count@\expandafter{\the\dimen\count@}%
2357         \advance\dimen@ \dimen\count@}%
2358     }
2359     {%
2360         \@for\x:=#2\do{%
2361             \advance \count@ \@ne
2362             \toks\count@\expandafter{\x\p@}%
2363             \advance\dimen@ \x\p@
2364             \dimen\count@ -\@ne\p@}%
2365         }%
2366         \advance\count@\@ne \dimen\itf@freespace=\count@
2367         \advance\count@\@ne \dimen\itf@tempdim=\count@
2368         \advance\count@\@ne \dimen\itf@sum=\count@
2369         \count@\z@
2370         \itf@freespace\z@ % free space left when  $w_i < x_i / \text{sum}(x_i) * \text{total}$ 
2371         \@for\x:=#3\do{%
2372             \advance \count@ \@ne
2373             \ifdim \dimen\count@<\z@ % not measured before
2374                 \setbox\z@=\hbox{\ignorespaces\x\unskip}%
2375                 \dimen\count@ = \wd\z@
2376             \fi
2377             \itf@tempdim = \itf@scaledimen{#1}*\the\toks\count@/\dimen@\relax
2378             \ifdim \dimen\count@ > \itf@tempdim
2379                 \dimen\count@ = \itf@tempdim
2380                 \edef\itf@istruncated{\itf@istruncated,\the\count@}%
2381                 \advance\itf@sum by\the\toks\count@
2382             \else
2383                 \advance\itf@freespace \itf@tempdim
2384                 \advance\itf@freespace -\dimen\count@
2385             \fi
2386             \edef\itf@trunc@width{\itf@trunc@width\the\count@\the\dimen\the\count@}}%
2387         }%
2388 %% Free space=\the\itf@freespace\par
2389 \ifdim\itf@freespace > \z@ % reinsert free space proportionally
2390     \ifx\itf@istruncated@gobble\else
2391         \@for\x:=\itf@istruncated\do{%
2392             \advance\dimen\x by\itf@scaledimen\itf@freespace*\the\toks\x/\itf@sum\relax
2393         }%
2394     \fi\fi
2395     \def\the@dimen{\the\dimen}%
2396     \edef\itf@trunc@width{\itf@trunc@width}%
2397     \expandafter\endgroup
2398     \expandafter\def
2399         \expandafter\itf@trunc@width
2400         \expandafter{\itf@trunc@width}%
2401 %% total=\the\dimexpr#1\relax
2402 }% \itf@htruncadjust
2403 \def\itf@scalefactor#1#2{\strip@pt\dimexpr#1*65536/#2\relax}
2404 \def\itf@scaledimen#1*#2/#3\relax{%
2405     \dimexpr\itf@scalefactor{#2}{#3}\dimexpr#1\relax\relax
2406 }% \itf@scaledimen

```

\htruncwidth

```

2407 \newcommand*\htruncwidth[1]{% purely expandable
2408     \expandafter\itf@htruncwidth\itf@trunc@width\relax\relax\@nnil{#1}%
2409 }% \htruncwidth
2410 \def\itf@htruncwidth#1#2#3\@nnil#4{%
2411     \ifx\relax#1\itf@trunc@nosuchwidth{#4}%
2412     \else\ifnum#1=#4
2413         #2%

```

```

2414 \else
2415 \itf@htruncwidth#3\@nnil{#4}%
2416 \fi\fi
2417}% \itf@htruncwidth
2418\def\itf@trunc@nosuchwidth#1{%
2419 \PackageError{interfaces-truncate}
2420 {You requested the width for text nr#1
2421 \MessageBreak but there is no such text
2422 \MessageBreak Therefore no such width!}\@ehd
2423}% \itf@trunc@nosuchwidth

2424 </truncate>

```

## 21.15 Interface **appendix** and the appendices environment

```

2425 <*appendix>
2426 \ProvidesPackage{interfaces-appendix}
2427 [2010/10/27 v1.2 easy control of appendices environment (FC)]
2428 \RequirePackage{interfaces-base}
2429 \itf@Check@Package{appendix}
2430 \itf@pgfkeys

```

\appendices

```

2431 \let\itf@appendices@pp\appendices
2432 \let\itf@endappendices@pp@endappendices
2433 \let\itf@appendices@pp\appendices
2434 \let\itf@endappendices@pp@endappendices
2435 \renewenvironment{appendices}[1][]{%
2436 \pgfqkeys{/itf/appendix/appendices}{#1}%
2437 \itf@appendices@pp
2438 }
2439 {\itf@endappendices@pp}%

2440 \pgfqkeys{/itf/appendix/appendices}{%
2441 title/.store as code in cs=appendixpagename,%
2442 title/.append code=\ifx\appendixpagename\@empty \@dopage@ppfalse
2443 \else \@dopage@pptrue\fi,%
2444 title/.value required,%
2445 intoc/.is if=@dotoc@pp,%
2446 intoc/title/.store as code in cs=appendixtocname,%
2447 intoc/title/.append code=\ifx\appendixtocname\@empty \@dotoc@ppfalse
2448 \else \@dotoc@pptrue\fi,%
2449 intoc/title/.default=\pgfkeysvalueof{/itf/appendix/appendices/title},%
2450 section/.is enum={section,chapter},%
2451 section/.append code={%
2452 \ifcase\pgfkeysvalueof{/itf/appendix/appendices/section/.number}\relax
2453 \@chapter@ppfalse\or\@chapter@pptrue\fi
2454 \if@chapter@pp\ifundef\chapter{%
2455 \PackageError{interfaces-appendix}
2456 {You asked chapter-style appendices but your
2457 \MessageBreak current class does not provide chapters!}\@eha
2458 \pgfkeysalso{/itf/appendix/appendices/section}}}%
2459 \fi},%
2460 chapter/.style={section=chapter},%
2461 chapter/.value forbidden,%
2462}% keys for appendices environment

2463 </appendix>

```

**21.16  Interface `\embedthisfile`**

```

2464 <*embedfile>
2465 \ProvidesPackage{interfaces-embedfile}
2466   [2010/10/27 v1.2 easy macros to embed files into pdf (FC)]
2467 \RequirePackage{interfaces-base}
2468 \itf@Check@Package{embedfile}
2469 \newrobustcmd*\embedthisfile{\@ifstar
2470   \embedthisfile@main
2471   \embedthisfile@all
2472 }% \embedthisfile
2473 \@onlypreamble\embedthisfile
2474 \def\embedthisfile@main{\embedfile{\jobname.tex}}
2475 \def\embedthisfile@all{%
2476   \embedthisfile@main
2477   \AfterPreamble{%
2478     \ltx@ifundefined{@listfiles}
2479       {\let\@listfiles\empty}% to avoid: \global\let\@filelist\relax in \document
2480       }%
2481   }%
2482   \gpreto\@enddocumenthook{%
2483     \let\do\itf@EmbedOne
2484     \expandafter\docsvlist\expandafter{\@filelist}%
2485   }%
2486 }% \embedthisfile@all
2487 \def\itf@EmbedOne#1{\itf@Embed@ne#1.tex\@nil}
2488 \def\itf@Embed@ne#1.tex#2\@nil{%
2489   \ltx@ifundefined{cp@#1}
2490     {}
2491     {\ifblank{#2}
2492       {}
2493       {\embedfile{#1.tex}}}%
2494 }% \itf@Embed@ne
2495 </embedfile>

```

**21.17  Interface `\envirom`**

```

2496 <*envirom>
2497 \ProvidesPackage{interfaces-envirom}
2498   [2010/07/23 v1.0 ReNewEnvirom = RenewEnvirom (FC)]
2499 \RequirePackage{interfaces-base}
2500 \AfterPackage{envirom}{
2501   \ifundef\ReNewEnvirom
2502     {\let\ReNewEnvirom\RenewEnvirom}
2503     }%
2504 }%
2505 </envirom>

```

**21.18  Interface `umrand`**

```

2506 <*umrand>
2507 \ProvidesPackage{interfaces-umrand}
2508   [2010/07/23 v1.0 patch for umrand.sty ifnotempty (FC)]
2509 \RequirePackage{interfaces-base}
2510 \BeforePackage{umrand}{%
2511   \let\itf@newif\newif
2512   \def\itf@tp{\ifnotempty}%
2513   \def\newif#1{\def\x{#1}\ifx\x\itf@tp \expandafter\@gobble
2514     \else\expandafter\ltx@newif\fi{#1}}%
2515 }
2516 \AfterPackage{umrand}{%
2517   \let\newif\itf@newif
2518   \let\itf@newif\@undefined
2519 }
2520 </umrand>

```

**21.19  Interface  `scrfile`**

```

2521 <*scrfile>
2522 \ProvidesPackage{interfaces-scrfile}
2523   [2010/10/27 v1.2 scrfile hooks improved (FC)]
2524 \RequirePackage{interfaces-base}

```

`\@pkgextension` In  $\text{\LaTeX}$  they are defined `\@onlypreamble`. So we use the `ltxcmds` equivalents:  
`\@clsextension`

```

2525 \AfterEndPreamble{%
2526   \let\@pkgextension\ltx@pkgextension
2527   \let\@clsextension\ltx@clsextension
2528   \let\@ifpackageloaded\ltx@ifpackageloaded
2529   \let\@ifclassloaded\ltx@ifclassloaded
2530 }

```

`\scr@load@hook`

```

2531 \renewrobustcmd*\scr@load@hook[2]{%
2532   \csuse{#2-@#1}%
2533   \ifcsname#2-@#1-\string\keep\endcsname\else\global\csundef{#2-@#1}\fi
2534 }% \scr@load@hook

```

`\scr@set@load@hook` (optimisation)

```

2535 \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`

```

2536 \renewrobustcmd*\BeforeFile{\@ifstar
2537   {\itf@s@LoadFile{before}}
2538   {\itf@LoadFile{before}}}%
2539 }% \BeforeFile
2540 \renewrobustcmd*\AfterFile{\@ifstar
2541   {\itf@s@LoadFile{after}}
2542   {\itf@LoadFile{after}}}%
2543 }% \AfterFile

2544 \def\itf@s@LoadFile#1#2{%
2545   \global\cslet{#2-@#1-\string\keep}\@empty
2546   \scr@set@load@hook{#1}{#2}%
2547 }% \itf@s@LoadFile
2548 \def\itf@LoadFile#1#2{%
2549   \global\csundef{#2-@#1-\string\keep}%
2550   \scr@set@load@hook{#1}{#2}%
2551 }% \itf@LoadFile

```

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

```

2552 \renewcommand*\@input[1]{\IfFileExists{#1}
2553   {\scr@load@hook{before}{#1}%
2554     \@input\@filef@und
2555     \scr@load@hook{after}{#1}}%
2556   {\typeout{No file #1.}}%
2557 }% \@input

```

```

2558 </scrfile>

```

## 21.20 † Interface pgfkeys (📖pgfmanual)

### 21.20.1 interfaces-pgfkeys

```

2559 \*pgfkeys>
2560 \ProvidesPackage{interfaces-pgfkeys}
2561 [2010/11/21 v2.3 pgfkeys handlers definitions (FC)]
2562 \RequirePackage{interfaces-base}
2563 \ifdefined\pgfkeys@ifexecutehandler % pgf version 2.1
2564 \def\itf@pgfkeys{%
2565   \global\let\pgfkeys@unpack      \itf@pgfkeys@unpack
2566   \global\let\pgfkeys@case@three  \itf@pgfkeys@case@three
2567   \global\let\pgfkeys@case@three@handleall\itf@pgfkeys@case@three@handleall
2568 }% \itf@pgfkeys
2569 \else
2570 \def\itf@pgfkeys{%
2571   \global\let\pgfkeys@unpack      \itf@pgfkeys@unpack
2572   \global\let\pgfkeys@case@three  \itf@pgfkeys@case@three
2573   \global\let\pgfkeys@addvalue    \itf@pgfkeys@addvalue
2574   \global\let\pgfkeys@exp@call    \itf@pgfkeys@exp@call
2575 }% \itf@pgfkeys
2576 \fi
2577 \AtBeginDocument{%
2578   \itf@pgfkeys
2579   \let\itf@pgfkeys\relax
2580 }% AtBeginDocument

```

#### Some new pgfkeys macros

`\pgfkeysmeaning`

```

2581 \providecommand*\pgfkeysmeaning[1]{%
2582   \pgfkeysifdefined{#1}
2583   {\expandafter\meaning\csname pgfk@#1\endcsname}
2584   {\meaning\@undefined}%
2585 }%

```

`\pgfkeysvalueof@unexpanded`

```

2586 \def\pgfkeysvalueof@unexpanded#1{%
2587   \unexpanded\expandafter\expandafter\expandafter{%
2588     \csname pgfk@#1\endcsname}%
2589 }% \pgfkeysvalueof@unexpanded

```

#### Some pgfkeys macros redefinition/corrections

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

```

2590 \ifdefined\pgfkeys@ifexecutehandler \else % pgf v2.1: bug fixed
2591 \def\itf@pgfkeysaddvalue#1#2#3{% (pgf v2.0: bug in pgfkeys.code.tex)
2592   \edef\pgfkeys@global@temp{%
2593     \unexpanded{#2}%
2594     \ifcsname pgfk@#1\endcsname
2595       \unexpanded\expandafter\expandafter\expandafter{%
2596         \csname pgfk@#1\endcsname}%
2597   \fi
2598   \unexpanded{#3}}%
2599   \pgfkeyslet{#1}\pgfkeys@global@temp
2600 }% \itf@pgfkeysaddvalue
2601 \fi

```

`\pgfkeysEsetvalue`  
`\pgfkeysEaddvalue`

```

2602 \ifdefined\pgfkeys@ifexecutehandler \long \fi % pgf version 2.1
2603 \def\pgfkeysEsetvalue#1#2{%
2604   \expandafter\edef\csname pgfk@#1\endcsname{#2}%
2605 }% \pgfkeysEsetvalue
2606 \ifdefined\pgfkeys@ifexecutehandler \long \fi % pgf v2.1
2607 \def\pgfkeysEaddvalue#1#2#3{% \csepreappto{pgfk@#1}{#2}{#3}
2608   \edef\pgfkeys@global@temp{%
2609     #2%
2610     \ifcsname pgfk@#1\endcsname
2611       \unexpanded\expandafter\expandafter\expandafter{%
2612         \csname pgfk@#1\endcsname}%
2613     \fi
2614     #3}%
2615   \pgfkeyslet{#1}\pgfkeys@global@temp
2616 }% \pgfkeysEaddvalue

```

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

```

2617%% missed a pair of braces around the argument in pgfkeys.code.tex (pgf v2.0)
2618%% ( the code could break if it contains = or , )
2619\ifdefined\pgfkeys@ifexecutehandler\else % (bug corrected in pgf v2.1)
2620  \def\itf@pgfkeys@exp@call#1{\pgfkeysalso{\pgfkeyscurrentpath={#1}}}%
2621 \fi

```

## Handlers `/.blank`

```

2622%% The value is now checked by \itf@pgfkeys@default@blank@required:
2623 \ifdefined\pgfkeys@ifexecutehandler \long \fi % pgf v2.1
2624 \def\itf@pgfkeys@unpack#1=#2=#3\pgfkeys@stop{%
2625   \pgfkeys@spdef\pgfkeyscurrentkey{#1}%
2626   \edef\pgfkeyscurrentkey{\pgfkeyscurrentkey}%
2627   \ifx\pgfkeyscurrentkey\pgfkeys@empty
2628     % Skip
2629   \else%
2630     \pgfkeys@add@path@as@needed
2631     \pgfkeys@spdef\pgfkeyscurrentvalue{#2}%
2632     \itf@pgfkeys@default@blank@required % <FC>
2633                                     \pgfkeyscurrentkey % <FC>
2634                                     \pgfkeys@case@one % <FC>
2635   \fi
2636 }% \itf@pgfkeys@unpack

```

`\itf@pgfkeys@default@blank@required` Handle the case: `/.@def` → `/.@blank` → `/.@def` ...

```

2637 \def\itf@pgfkeys@default@blank@required#1#2{%
2638   \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text% Hmm... no value
2639     \pgfkeysifdefined{#1/.@def}%
2640     {\pgfkeysgetvalue{#1/.@def}\pgfkeyscurrentvalue}
2641     {}% no default, so leave it
2642   \fi
2643   \ifx\pgfkeyscurrentvalue\pgfkeys@empty
2644     \pgfkeysifdefined{#1/.@blank}
2645     {\pgfkeysgetvalue{#1/.@blank}\pgfkeyscurrentvalue}
2646     {}% no /.blank specified
2647   \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text% in case /.blank -> \pgfkeysnovalue
2648     \pgfkeysifdefined{#1/.@def}%
2649     {\pgfkeysgetvalue{#1/.@def}\pgfkeyscurrentvalue}
2650     {}% no default, so leave it
2651   \fi\fi
2652   \ifx\pgfkeyscurrentvalue\pgfkeysvaluerequired
2653     \pgfkeysvalueof{/errors/value required/.@cmd}\pgfkeyscurrentkey
2654                                     \pgfkeyscurrentvalue

```

```

2655                                     \pgfeov
2656     \else
2657         #2%
2658     \fi
2659}% \itf@pgfkeys@default@blank@required

```

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

```

2660%% To handle the cases of: /handler/.default= /handler/.blank= /handler/.value required,
2661%% it is necessary to redefine \pgfkeys@case@three
2662\def\itf@pgfkeys@case@three{%
2663    \pgfkeys@split@path
2664    \pgfkeysifdefined{/handlers/\pgfkeyscurrentname/.@cmd}{
2665        {%
2666            \itf@pgfkeys@default@blank@required{/handlers/\pgfkeyscurrentname}%
2667            {\pgfkeysgetvalue{/handlers/\pgfkeyscurrentname/.@cmd}\pgfkeys@code
2668             \expandafter\pgfkeys@code\pgfkeyscurrentvalue\pgfeov}%
2669        }
2670        \pgfkeys@unknown % no handler, thus the key is unknown
2671}% \itf@pgfkeys@case@three
2672\ifdefined\pgfkeys@ifexecutehandler
2673    \let\itf@pgfkeys@case@three@handleall \itf@pgfkeys@case@three
2674\fi

```

### Loading interfaces modifications for `pgfkeys`

```

2675\itf@pgfkeys

```

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

```

2676\pgfkeys{%
2677    /handlers/.alias/.code=\itf@pgfkeys@ifaliasexists{#1}{%
2678        \pgfkeysedef\pgfkeyscurrentpath{%
2679            \noexpand\pgfkeysalso{\pgfkeysdefaultpath#1={##1}}},%
2680        /handlers/.alias/.value required,%
2681        /handlers/.blank/.code=\pgfkeyssetvalue{\pgfkeyscurrentpath/.@blank}{#1},%
2682        /handlers/.blank/.default=\pgfkeysnovalue,%
2683}% /.alias and /.blank
2684\def\itf@pgfkeys@ifaliasexists#1#2{%
2685    \pgfkeysifdefined{\pgfkeysdefaultpath#1/.@cmd}{#2}
2686    {\pgfkeysifdefined{\pgfkeysdefaultpath#1}
2687     {#2}
2688     {\itf@pgfkeys@aliaserror{#1}}}%
2689}% \itf@pgfkeys@ifaliasexists
2690\def\itf@pgfkeys@aliaserror#1{%
2691    \PackageError{interfaces-pgfkeys}
2692        {You want the key \pgfkeyscurrentpath\space
2693         \MessageBreak to be an alias for \pgfkeysdefaultpath#1
2694         \MessageBreak but the key \pgfkeysdefaultpath#1
2695         \MessageBreak is not defined!}\@eha\@gobble
2696}% \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).

```

2697 \pgfkeys{
2698   /handlers/.alias*/.code=\itf@pgfkeys@ifaliasexists{#1}{%
2699     \pgfkeysdef\pgfkeyscurrentpath{\noexpand\pgfkeysalso{\pgfkeysdefaultpath#1={##1}}}%
2700     \@tfor\x:=+-*\do{%
2701       \pgfkeysifdefined{\pgfkeysdefaultpath#1\x/.@cmd}
2702       {\pgfkeysdef{\pgfkeyscurrentpath\x}{%
2703         \noexpand\pgfkeysalso{\pgfkeysdefaultpath#1\x={##1}}}}
2704       {}}%
2705     },%
2706   /handlers/.alias*/.value required,%
2707 }% /.alias*
```

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

```

2708 \pgfkeys{%
2709   /handlers/.is private/.code=\pgfkeysdef\pgfkeyscurrentpath{%
2710     \PackageError{#1}{Key \pgfkeyscurrentkey\space is private
2711     \MessageBreak and should not be modified}\@ehd},%
2712   /handlers/.is private to class/.code=\pgfkeysdef\pgfkeyscurrentpath{%
2713     \ClassError{#1}{Key \pgfkeyscurrentkey\space is private
2714     \MessageBreak and should not be modified}\@ehd}%
2715 }% /.is private
```

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

```

2716 \def\itf@pgfkeys@choice[#1]#2#3{%
2717   \pgfkeys{\pgfkeyscurrentpath/.cd,%
2718     .is choice,%
2719     #2/.code={#3},%
2720     .default/.expanded=\@firstoftwo#1}%
2721 }% \itf@pgfkeys@choice
2722 \pgfkeys{%
2723   /handlers/.true/.code=\itf@pgfkeys@choice[{true}]{false}[{true}]{#1},%
2724   /handlers/.false/.code=\itf@pgfkeys@choice[{true}]{false}[{false}]{#1},%
2725   /handlers/.on/.code=\itf@pgfkeys@choice[{on}]{off}[{on}]{#1},%
2726   /handlers/.off/.code=\itf@pgfkeys@choice[{on}]{off}[{off}]{#1},%
2727 }% /.true /.false
```

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

```

2728 \pgfkeys{%
2729   /handlers/.store in cs/.code=\pgfkeysdef\pgfkeyscurrentpath{\csdef{#1}{##1}},%
2730   /handlers/.estore in cs/.code=\pgfkeysdef\pgfkeyscurrentpath{\csedef{#1}{##1}},%
2731 }% /.store in cs

```

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

```

2732 %% /.store as code handlers
2733 \pgfkeys{%
2734   /handlers/.store as code/.code={%
2735     \pgfkeysedef\pgfkeyscurrentpath{%
2736       \noexpand\pgfkeyslet{\pgfkeyscurrentpath}\noexpand\pgfkeyscurrentvalue}%
2737     \pgfkeyslet\pgfkeyscurrentpath\@empty
2738     \pgfkeyslet{\pgfkeyscurrentpath/.@def}\@empty
2739     \@tfor\x:=#1\do{%
2740       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2741         \noexpand\pgfkeysalso{\pgfkeyscurrentpath/.expanded={%
2742           \noexpand\pgfkeysvalueof@unexpanded{\pgfkeyscurrentpath}%
2743           \noexpand\unexpanded{##1}}}%
2744         \pgfkeyslet{\pgfkeyscurrentpath\x/.@def}\@empty
2745       }},%
2746   /handlers/.store as code/.default=*,%
2747 }% /.store as code

```

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

```

2748 \pgfkeys{%
2749   /handlers/.estore as code/.code={%
2750     \pgfkeysedef\pgfkeyscurrentpath{%
2751       \noexpand\pgfkeysEsetvalue{\pgfkeyscurrentpath}{##1}}%
2752     \pgfkeyslet\pgfkeyscurrentpath\@empty
2753     \@tfor\x:=#1\do{%
2754       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2755         \noexpand\pgfkeysalso{\pgfkeyscurrentpath/.expanded={%
2756           \noexpand\pgfkeysvalueof@unexpanded{\pgfkeyscurrentpath}%
2757           \noexpand\unexpanded{##1}}}%
2758       }},%
2759   /handlers/.estore as code/.default=*,%
2760 }% /.estore as code

```

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

```

2761 \pgfkeys{%
2762   /handlers/.store as code in cs/.code={%
2763     \pgfkeysdef\pgfkeyscurrentpath{%
2764       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
2765       \else\csdef{#1}{##1}\fi}%
2766     \@tfor\x:=*+\do{%
2767       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2768         \noexpand\pgfkeysalso{\pgfkeyscurrentpath/.expanded={%
2769           \noexpand\ifcsname#1\endcsname
2770           \noexpand\expandafter\noexpand\expandonce
2771           \noexpand\csname#1\endcsname
2772           \noexpand\fi
2773           \noexpand\unexpanded{##1}}}%
2774         \pgfkeyslet{\pgfkeyscurrentpath\x/.@def}\@empty
2775       }},%
2776   /handlers/.store as code in cs/.value required,%
2777 }% /.store as code in cs
2778 \pgfkeys{%
2779   /handlers/.estore as code in cs/.code={%

```

```

2780 \pgfkeysdef\pgfkeyscurrentpath{%
2781 \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
2782 \else\csedef{##1}{##1}\fi}%
2783 \@tfor\x:=*+\do{%
2784 \pgfkeysedef\pgfkeyscurrentpath\x}{%
2785 \noexpand\pgfkeysalso{\pgfkeyscurrentpath/.expanded=%
2786 \noexpand\ifcsname#1\endcsname
2787 \noexpand\expandafter\noexpand\expandonce
2788 \noexpand\csname#1\endcsname
2789 \noexpand\fi
2790 \noexpand\unexpanded{##1}}}%
2791 }},% default if \pgfkeysnovalue which expands to \@empty
2792 /handlers/.estore as code in cs/.value required,%
2793 }% /.estore as code in cs

```

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

```

2794 \protected\def\itf@pgfk@color#1#2\@nil{%
2795 \ifx\@let@token\color
2796 \def\pgfkeyscurrentvalue{#2}%
2797 \else\ifx\@let@token\normalcolor
2798 \def\pgfkeyscurrentvalue{default}%
2799 \else\ifx\@let@token\pgfkeysnovalue
2800 \def\pgfkeyscurrentvalue{default}%
2801 \fi\fi\fi
2802 }% \itf@pgfk@color
2803 \pgfkeys{%
2804 /handlers/.store as color/.code=%
2805 \pgfkeysedef\pgfkeyscurrentpath{%
2806 \futurelet\noexpand\@let@token\itf@pgfk@color
2807 ##1\noexpand\pgfkeysnovalue\noexpand\@nil
2808 \noexpand\pgfkeyslet{\pgfkeyscurrentpath}\noexpand\pgfkeyscurrentvalue}%
2809 }% /.store as color

```

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

```

2810 \pgfkeys{%
2811 /handlers/.store as color in cs/.code=%
2812 \pgfkeysedef\pgfkeyscurrentpath{%
2813 \futurelet\noexpand\@let@token\itf@pgfk@color
2814 ##1\noexpand\pgfkeysnovalue\noexpand\@nil
2815 \cslet{##1}\pgfkeyscurrentvalue}%
2816 }% /.store as color in cs

```

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

```

2817 \protected\def\itf@pgfk@colorcmd#1\@nil{%
2818 \ifcat @\noexpand\@let@token
2819 \def\pgfkeyscurrentvalue{\color{##1}}%
2820 \else\ifcat 0\noexpand\@let@token
2821 \def\pgfkeyscurrentvalue{\color{##1}}%
2822 \else\ifx\@let@token\@nil
2823 \let\pgfkeyscurrentvalue\normalcolor
2824 \fi\fi\fi
2825 }% \itf@pgfk@colorcmd
2826 \pgfkeys{%
2827 /handlers/.store as color command/.code=%
2828 \pgfkeysedef\pgfkeyscurrentpath{%
2829 \futurelet\noexpand\@let@token\itf@pgfk@colorcmd
2830 ##1\noexpand\@nil
2831 \noexpand\pgfkeyslet{\pgfkeyscurrentpath}\noexpand\pgfkeyscurrentvalue}%
2832 \pgfkeyssetvalue{\pgfkeyscurrentpath/.def}\normalcolor
2833 }% /.store as color command

```

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

```

2834 \pgfkeys{%
2835   /handlers/.store as color command in cs/.code=%
2836   \pgfkeysedef\pgfkeyscurrentpath{%
2837     \futurelet\noexpand\@let@token\itf@pgfk@colorcmd
2838     ##1\noexpand\@nil
2839     \cslet{##1}\noexpand\pgfkeyscurrentvalue}%
2840   \pgfkeyssetvalue{\pgfkeyscurrentpath/.@def}\normalcolor
2841 }% /.store as color command in cs

```

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

```

2842 \pgfkeys{%
2843   /handlers/.store as number/.code={%
2844     \pgfkeysedef\pgfkeyscurrentpath{%
2845       \noexpand\pgfkeyssetvalue{\pgfkeyscurrentpath}{\numexpr##1\relax}}%
2846     \pgfkeyslet\pgfkeyscurrentpath\z@
2847     \pgfkeyslet{\pgfkeyscurrentpath/.@def}\z@
2848     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@
2849     \@tfor\x:=#1\do{%
2850       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2851         \noexpand\pgfkeysalso{\pgfkeyscurrentpath/.expanded=%
2852           \noexpand\pgfkeysvalueof@unexpanded{\pgfkeyscurrentpath}%
2853           \x
2854           \numexpr\unexpanded{##1}\relax}}%
2855       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x\@ne\else\z@\fi}%
2856       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x\@ne\else\z@\fi}%
2857     }},%
2858   /handlers/.store as number/.default=+-*,%
2859 }% /.store as number

```

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

```

2860 \pgfkeys{%
2861   /handlers/.estore as number/.code={%
2862     \pgfkeysedef\pgfkeyscurrentpath{%
2863       \noexpand\pgfkeyssetvalue{\pgfkeyscurrentpath}{%
2864         \noexpand\number\numexpr##1\relax}}%
2865     \pgfkeyslet\pgfkeyscurrentpath\z@
2866     \pgfkeyslet{\pgfkeyscurrentpath/.@def}\z@
2867     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@
2868     \@tfor\x:=#1\do{%
2869       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2870         \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
2871           \noexpand\pgfkeysvalueof{\pgfkeyscurrentpath}%
2872           \x
2873           \numexpr##1\relax}}%
2874     }},%
2875   /handlers/.estore as number/.default=+-*,%
2876 }% /.estore as number

```

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

```

2877 \pgfkeys{%
2878   /handlers/.store as number in cs/.code={%
2879     \pgfkeysdef\pgfkeyscurrentpath{%
2880       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
2881       \else\csdef{#1}{\numexpr##1\relax}\fi}%
2882     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@
2883     \@tfor\x:=+-*\do{%
2884       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2885         \letcs\noexpand\pgfkeys@temp{%
2886           \noexpand\ifcsname#1\endcsname
2887             #1\noexpand\else\if*\x @ne\else z@\fi\noexpand\fi}%
2888         \edef\noexpand\pgfkeys@temp{%
2889           \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
2890             \noexpand\expandonce\noexpand\pgfkeys@temp
2891             \x
2892             \numexpr\noexpand\unexpanded{##1}\relax}}%
2893         }\noexpand\pgfkeys@temp}%
2894     \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else z@\fi}%
2895     \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else z@\fi}%
2896   }},%
2897   /handlers/.store as number in cs/.value required,%
2898 }% /.store as number in cs

```

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

```

2899 \pgfkeys{%
2900   /handlers/.estore as number in cs/.code={%
2901     \pgfkeysdef\pgfkeyscurrentpath{%
2902       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
2903       \else\csdef{#1}{\number\numexpr##1\relax}\fi}%
2904     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@
2905     \@tfor\x:=+-*\do{%
2906       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2907         \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
2908           \noexpand\csname\noexpand\ifcsname#1\endcsname
2909             #1\noexpand\else\if*\x @ne\else z@\fi\noexpand\fi\endcsname
2910           \x
2911           \numexpr##1\relax}}%
2912         \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else z@\fi}%
2913         \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else z@\fi}%
2914       }},%
2915     /handlers/.estore as number in cs/.value required,%
2916 }% /.estore as number in cs

```

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

```

2917 \pgfkeys{%
2918   /handlers/.store in counter cs/.code={%
2919     \pgfkeysdef\pgfkeyscurrentpath{%
2920       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
2921       \else\csname#1\endcsname\numexpr##1\relax\fi}%
2922     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@
2923     \@tfor\x:=+-*\do{%
2924       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
2925         \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
2926           \noexpand\csname\noexpand\ifcsname#1\endcsname
2927             #1\noexpand\else\if*\x @ne\else z@\fi\noexpand\fi\endcsname
2928           \x
2929           \numexpr##1\relax}}%
2930         \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else z@\fi}%
2931         \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else z@\fi}%
2932       }},%

```

```

2933 /handlers/.store in counter cs/.value required,%
2934 }% /.store in counter cs

```

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

```

2935 \pgfkeys{%
2936   /handlers/.store as glue/.code={%
2937     \pgfkeysdef\pgfkeyscurrentpath{%
2938       \noexpand\pgfkeyssetvalue{\pgfkeyscurrentpath}{\glueexpr##1\relax}}%
2939     \pgfkeyslet\pgfkeyscurrentpath\z@skip
2940     \pgfkeyslet{\pgfkeyscurrentpath/.@def}\z@skip
2941     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@skip
2942     \@tfor\x:=#1\do{%
2943       \pgfkeysdef{\pgfkeyscurrentpath\x}{%
2944         \noexpand\pgfkeysalso{\pgfkeyscurrentpath/.expanded=%
2945           \noexpand\pgfkeysvalueof@unexpanded{\pgfkeyscurrentpath}%
2946           \x
2947           \if*\x\noexpand\itf@units@scale##1!
2948           \else\glueexpr\noexpand\unexpanded{##1\relax}\fi}}%
2949       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else\z@skip\fi}%
2950       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else\z@skip\fi}%
2951     }},%
2952   /handlers/.store as glue/.default=+-*,%
2953 }% /.store as glue

```

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

```

2954 \pgfkeys{%
2955   /handlers/.estore as glue/.code={%
2956     \pgfkeysdef{\pgfkeyscurrentpath}{%
2957       \noexpand\pgfkeysEsetvalue{\pgfkeyscurrentpath}{%
2958         \noexpand\the\glueexpr##1\relax}}%
2959     \pgfkeyslet\pgfkeyscurrentpath\z@skip
2960     \pgfkeyslet{\pgfkeyscurrentpath/.@def}\z@skip
2961     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@skip
2962     \@tfor\x:=#1\do{%
2963       \pgfkeysdef{\pgfkeyscurrentpath\x}{%
2964         \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
2965           \noexpand\pgfkeysvalueof{\pgfkeyscurrentpath}\x
2966           \if*\x\noexpand\itf@units@scale##1!
2967           \else\glueexpr##1\relax\fi}}%
2968       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else\z@skip\fi}%
2969       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else\z@skip\fi}%
2970     }},%
2971   /handlers/.estore as glue/.default=+-*,%
2972 }% /.estore as glue

```

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

```

2973 \pgfkeys{%
2974   /handlers/.store as dimen/.code={%
2975     \pgfkeysdef\pgfkeyscurrentpath{%
2976       \noexpand\pgfkeyssetvalue{\pgfkeyscurrentpath}{\dimexpr##1\relax}}%
2977     \pgfkeyslet\pgfkeyscurrentpath\z@
2978     \pgfkeyslet{\pgfkeyscurrentpath/.@def}\z@
2979     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@
2980     \@tfor\x:=#1\do{%
2981       \pgfkeysdef{\pgfkeyscurrentpath\x}{%
2982         \noexpand\pgfkeysalso{\pgfkeyscurrentpath/.expanded={%
2983           \noexpand\pgfkeysvalueof@unexpanded{\pgfkeyscurrentpath}%
2984           \x
2985           \if*\x\noexpand\itf@units@scale##1!

```

```

2986             \else\dimexpr\noexpand\unexpanded{##1}\relax\fi}}}%
2987     \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*x\@ne\else\z@\fi}%
2988     \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*x\@ne\else\z@\fi}%
2989     }},%
2990     /handlers/.store as dimen/.default=+-*,%
2991 }% /.store as dimen

```

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

```

2992 \pgfkeys{%
2993   /handlers/.estore as dimen/.code={%
2994     \pgfkeysedef\pgfkeyscurrentpath{%
2995       \noexpand\pgfkeysEsetvalue{\pgfkeyscurrentpath}{%
2996         \noexpand\the\dimexpr##1\relax}}}%
2997     \pgfkeyslet\pgfkeyscurrentpath\z@
2998     \pgfkeyslet{\pgfkeyscurrentpath/.@def}\z@
2999     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@
3000     \@tfor\x:=#1\do{%
3001       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
3002         \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
3003           \noexpand\pgfkeysvalueof{\pgfkeyscurrentpath}\x
3004           \if*x\@noexpand\itf@units@scale##1!
3005           \else\dimexpr##1\relax\fi}}}%
3006       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*x\@ne\else\z@\fi}%
3007       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*x\@ne\else\z@\fi}%
3008     }},%
3009     /handlers/.estore as dimen/.default=+-*,%
3010 }% /.estore as dimen

```

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

```

3011 \pgfkeys{%
3012   /handlers/.store as glue in cs/.code={%
3013     \pgfkeysdef\pgfkeyscurrentpath{%
3014       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
3015       \else\csdef{##1}{\glueexpr##1\relax}\fi}%
3016     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@skip
3017     \@tfor\x:=+-*\do{%
3018       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
3019         \letcs\noexpand\pgfkeys@temp{%
3020           \noexpand\ifcsname#1\endcsname
3021           #1\noexpand\else\z@skip\noexpand\fi}%
3022         \edef\noexpand\pgfkeys@temp{%
3023           \noexpand\pgfkeysalso{\pgfkeyscurrentpath={%
3024             \noexpand\expandonce\noexpand\pgfkeys@temp
3025             \x
3026             \if*x\noexpand\itf@units@scale##1!
3027             \else\glueexpr\noexpand\unexpanded{##1}\relax\fi}}}%
3028         }\noexpand\pgfkeys@temp}%
3029       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*x\@ne\else\z@skip\fi}%
3030       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*x\@ne\else\z@skip\fi}%
3031     }},%
3032     /handlers/.store as glue in cs/.value required,%
3033 }% /.store as glue in cs

```

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

```

3034 \pgfkeys{%
3035   /handlers/.estore as glue in cs/.code={%
3036     \pgfkeysdef\pgfkeyscurrentpath{%
3037       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
3038       \else\csdef{#1}{\the\glueexpr##1\relax}\fi}%
3039     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@skip
3040     \@tfor\x:=+-*\do{%
3041       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
3042         \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
3043           \noexpand\csname\noexpand\ifcsname#1\endcsname
3044             #1\noexpand\else z@skip\noexpand\fi\endcsname
3045           \x
3046           \if*\x\noexpand\itf@units@scale##1!
3047           \else\glueexpr##1\relax\fi}}%
3048       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else z@skip\fi}%
3049       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else z@skip\fi}%
3050     }},%
3051   /handlers/.etore as glue in cs/.value required,%
3052 }% /.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` (*ie.* a length). The skip is given by its name, not as a control sequence.

```

3053 \pgfkeys{%
3054   /handlers/.store in skip cs/.code={%
3055     \pgfkeysdef\pgfkeyscurrentpath{%
3056       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
3057       \else\csname#1\endcsname\glueexpr##1\relax\fi}%
3058     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@skip
3059     \@tfor\x:=+-*\do{%
3060       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
3061         \noexpand\pgfkeysalso{\pgfkeyscurrentpath=%
3062           \csname#1\endcsname % should have been defined as a skip (not expandable)
3063           \x
3064           \if*\x\noexpand\itf@units@scale##1!
3065           \else\glueexpr##1\relax\fi}}%
3066       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*\x@ne\else z@skip\fi}%
3067       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*\x@ne\else z@skip\fi}%
3068     }},%
3069   /handlers/.store in skip cs/.value required,%
3070 }% /.store in skip cs

```

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

```

3071 \pgfkeys{%
3072   /handlers/.store as dimen in cs/.code={%
3073     \pgfkeysdef\pgfkeyscurrentpath{%
3074       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
3075       \else\csdef{#1}{\dimexpr##1\relax}\fi}%
3076     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@
3077     \@tfor\x:=+-*\do{%
3078       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
3079         \letcs\noexpand\pgfkeys@temp{%
3080           \ifcsname#1\endcsname#1\else z@\fi}%
3081         \edef\noexpand\pgfkeys@temp{%
3082           \noexpand\pgfkeysalso{\pgfkeyscurrentpath={%
3083             \noexpand\expandonce\noexpand\pgfkeys@temp
3084             \x
3085             \if*\x\noexpand\itf@units@scale##1!
3086             \else\dimexpr\noexpand\unexpanded{##1}\fi}}%
3087         }\noexpand\pgfkeys@temp}%

```

```

3088      \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*x\@ne\else\z@\fi}%
3089      \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*x\@ne\else\z@\fi}%
3090    }},%
3091    /handlers/.store as dimen in cs/.value required,%
3092}% /.store as dimen in cs {csname}{+-}

```

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

```

3093 \pgfkeys{%
3094   /handlers/.estore as dimen in cs/.code={%
3095     \pgfkeysdef\pgfkeyscurrentpath{%
3096       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
3097       \else \@defaultunits\itf@sk ##1pt\relax\@nnil
3098       \csedef{#1}{\the\dimexpr\itf@sk\relax}\fi}%
3099     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@
3100     \@tfor\x:=+~*\do{%
3101       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
3102         \noexpand\ifcsname#1\endcsname
3103         \noexpand\@defaultunits\itf@sk
3104         \noexpand\csname#1\endcsname pt\relax\noexpand\@nnil
3105         \noexpand\else\itf@sk\z@\noexpand\fi
3106         \if*x
3107         \itf@sk\dimexpr\itf@sk*\noexpand\itf@units@scale##1!\relax
3108         \else
3109         \noexpand\@defaultunits\@tempdima ##1pt\relax\noexpand\@nnil
3110         \advance\itf@sk\x\@tempdima
3111         \fi
3112         \noexpand\pgfkeysalso{\pgfkeyscurrentpath=\itf@sk}}%
3113       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*x\@ne\else\z@\fi}%
3114       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*x\@ne\else\z@\fi}%
3115     }},%
3116   /handlers/.etore as dimen in cs/.value required,%
3117}% /.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.

```

3118 \pgfkeys{%
3119   /handlers/.store in dimen cs/.code={%
3120     \pgfkeysdef\pgfkeyscurrentpath{%
3121       \ifx\pgfkeyscurrentvalue\pgfkeysnovalue@text
3122       \else\csname#1\endcsname\dimexpr##1\relax\fi}%
3123     \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\z@
3124     \@tfor\x:=+~*\do{%
3125       \pgfkeysedef{\pgfkeyscurrentpath\x}{%
3126         \if*x \else
3127         \noexpand\@defaultunits\itf@sk ##1pt\relax\noexpand\@nnil
3128         \fi
3129         \noexpand\pgfkeysalso{\pgfkeyscurrentpath=
3130           \csname#1\endcsname % has been defined as a dimen (not expandable)
3131           \x
3132           \if*x \noexpand\itf@units@scale##1!
3133           \else\itf@sk\fi}}%
3134       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@def}{\if*x\@ne\else\z@\fi}%
3135       \pgfkeysEsetvalue{\pgfkeyscurrentpath\x/.@blank}{\if*x\@ne\else\z@\fi}%
3136     }},%
3137   /handlers/.store in dimen cs/.value required,%
3138}% /.store in dimen cs

```

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

```

3139 \pgfkeys{%
3140   /handlers/.is multiple choice/.code=%
3141     \pgfkeys{\pgfkeyscurrentpath/.cd,%
3142       .unknown/.style={/errors/unknown choice value=\pgfkeyscurrentkey\pgfkeyscurrentvalue},%
3143       .code=\expandafter\pgfkeys\expandafter{\pgfkeyscurrentkey/.cd,###1}%
3144     },%
3145   /handlers/.is multiple choice/.value forbidden,%
3146}% /.is multiple choice

```

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

```

3147 \pgfkeys{%
3148   /handlers/.is enum/.code={%
3149     \itf@cnt\z@
3150     \@for\x:=\pgfkeyscurrentvalue\do{%
3151       \pgfkeysedef{\pgfkeyscurrentpath/\x}{%
3152         \noexpand\pgfkeyssetvalue{\pgfkeyscurrentpath}{\x}%
3153         \noexpand\pgfkeyssetvalue{\pgfkeyscurrentpath/.number}{\the\itf@cnt}}%
3154       \ifnum\itf@cnt=\z@
3155         \pgfkeyslet{\pgfkeyscurrentpath/.@def}\x
3156         \pgfkeyslet{\pgfkeyscurrentpath/.@blank}\x
3157       \fi
3158       \advance\itf@cnt\@ne
3159     }%
3160     \pgfkeysalso{\pgfkeyscurrentpath/.is choice}},%
3161   /handlers/.is enum/.value required,%
3162}% /.is enum

3163 \pgfkeys>

```

**21.21  Interface **etoolbox** - etoolbox-like commands**

```

3164 (*etoolbox)
3165 \ProvidesPackage{interfaces-etoolbox}
3166   [2010/10/27 v1.2 some more etoolbox-like commands (FC)]
3167 \RequirePackage{interfaces-base}
3168 \begingroup\catcode'\&=3
3169   \long\gdef\etb@doitem#1&\do{#1}}
3170 \endgroup

\preappto
\epreappto
\gpreappto
\xpreappto
3171 \providerobustcmd\preappto[3]{%
3172   \edef#1{\unexpanded{#2}\ifdefined#1\expandonce#1\fi\unexpanded{#3}}%
3173}% \preappto
3174 \providerobustcmd\epreappto[3]{%
3175   \edef#1{#2\ifdefined#1\expandonce#1\fi#3}%
3176}% \epreappto
3177 \providerobustcmd*\gpreappto{\global\preappto}
3178 \providerobustcmd*\xpreappto{\global\epreappto}

\cspreappto
\csepreappto
\csgpreappto
\csxpreappto
3179 \providerobustcmd*\cspreappto[1]{%
3180   \csname itf@\ifcsname#1\endcsname
3181     preappto\else defpreapp\fi\expandafter\endcsname
3182     \csname#1\endcsname
3183}% \cspreappto
3184 \long\def\itf@preappto#1#2#3{\edef#1{#2\expandonce#1#3}}
3185 \long\def\itf@defpreapp#1#2#3{\edef#1{#2#3}}
3186 \providerobustcmd*\csepreappto[1]{%
3187   \csname itf@\ifcsname#1\endcsname

```

```

3188      \preappto\else \def\preapp\fi\expandafter\endcsname
3189      \csname#1\endcsname
3190 }% \csepreappto
3191 \providerobustcmd*\csgpreappto{\global\cspreappto}
3192 \providerobustcmd*\csxpreappto{\global\csepreappto}

```

---

```

\removefrom
\eremovefrom
\gremovefrom
\xremovefrom
3193 \providerobustcmd\removefrom[2]{\in@false
3194   \long\def\itf@tp##1\itf@tp{%
3195     \in@true
3196     \edef#1{\unexpanded{##1}}%
3197   }\expandafter\itf@removefrom\expandafter{#1}{#2}%
3198 }% \removefrom
3199 \providerobustcmd\eremovefrom[2]{\begingroup
3200   \edef\x{\endgroup\removefrom\noexpand#1{#2}%
3201   }\x
3202 }% \eremovefrom
3203 \providerobustcmd\gremovefrom[2]{%
3204   \removefrom{#1}{#2}%
3205   \global\let#1=#1%
3206 }% \gremovefrom
3207 \providerobustcmd\xremovefrom[2]{%
3208   \eremovefrom{#1}{#2}%
3209   \global\let#1=#1%
3210 }% \xremovefrom

```

```

\itf@removefrom
3211 \long\def\itf@removefrom#1#2{% removes #2 (code) from #1 (hook)
3212   \def\in@##1#2##2\in@@{\itf@tp##1##2\in@@}%
3213   \def\in@@##1\in@@##2\in@@{%
3214     \ifx\in@##2\else##1\fi
3215   }\expandafter\in@@\in@@#1\itf@tp\in@@#2\in@\in@@
3216 }% \itf@removefrom

```

```

\removefromstart
\eremovefromstart
\gremovefromstart
\xremovefromstart
3217 \newrobustcmd*\removefromstart[2]{\in@false
3218   \def\itf@tp##1\itf@tp{%
3219     \in@true
3220     \edef#1{\unexpanded{##1}}%
3221   }\expandafter\itf@removefromstart\expandafter{#1}{#2}%
3222 }% \removefromstart
3223 \providerobustcmd*\eremovefromstart[2]{\begingroup
3224   \edef\x{\endgroup\removefromstart\noexpand#1{#2}%
3225   }\x
3226 }% \eremovefromstart
3227 \providerobustcmd*\gremovefromstart[2]{%
3228   \removefromstart{#1}{#2}%
3229   \global\let#1=#1%
3230 }% \gremovefromstart
3231 \providerobustcmd*\xremovefromstart[2]{%
3232   \eremovefromstart{#1}{#2}%
3233   \global\let#1=#1%
3234 }% \xremovefromstart

```

```

\itf@removefromstart

```

```

3235 \def\itf@removefromstart#1#2{% removes #2 (code) from #1 (hook)

```

```

3236 \def\in@@##1#2##2\in@@{##1\in@\itf@tp##2\in@@}%
3237 \def\in@@##1\in@@##2\in@@{%
3238 \ifx\in@@##1\fi
3239 }\expandafter\in@@\in@@#1\itf@tp\in@@#2\in@@
3240}% \itf@removefromstart

```

---

```

3241 </etoolbox>

```

## 22 History

### [2010/11/21 v2.3]

- `interfaces-fancyhdr` : new keys and new aliases for header, footer and offset settings

### [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 `\ttl@{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  $\LaTeX$ . `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  $\mathcal{E}$ -TeX `\marks` registers (useful with `\sectionformat` and `\pagesetup`).

`interfaces` is mainly based on `scrfile.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                                      |                                                                                                                                   |
|----------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| <code>\#</code> .....                        | 102, 110, 119, 689, 701                                                                                                           |
| <code>\&amp;</code> .....                    | 3168                                                                                                                              |
| <code>\@@input</code> .....                  | 2554                                                                                                                              |
| <code>\@addtoreset</code> .....              | 538, 546, 584, 586                                                                                                                |
| <code>\@break@tfor</code> .....              | 192                                                                                                                               |
| <code>\@cftmakeloftitle_(patch)</code> ..... | <u>1903</u>                                                                                                                       |
| <code>\@cftmakelottitle_(patch)</code> ..... | <u>1903</u>                                                                                                                       |
| <code>\@cftmaketoctitle_(patch)</code> ..... | <u>1903</u>                                                                                                                       |
| <code>\@cftpnumoff</code> .....              | 1757, 1885                                                                                                                        |
| <code>\@cftpnumon</code> .....               | 1756, 1884                                                                                                                        |
| <code>\@clsextension</code> .....            | <u>2525</u>                                                                                                                       |
| <code>\@defaultunits</code> .....            | 146, 147, 198, 213, 736, 3097, 3103, 3109, 3127                                                                                   |
| <code>\@elt</code> .....                     | 1004                                                                                                                              |
| <code>\@expandtwoargs</code> .....           | 192, 207, 208                                                                                                                     |
| <code>\@filef@und</code> .....               | 2554                                                                                                                              |
| <code>\@filelist</code> .....                | 2479, 2484                                                                                                                        |
| <code>\@firstofone</code> .....              | 2006, 2282                                                                                                                        |
| <code>\@fourthoffive</code> .....            | 928                                                                                                                               |
| <code>\@ifclassloaded</code> .....           | 2529                                                                                                                              |
| <code>\@ifpackageloaded</code> .....         | 2528                                                                                                                              |
| <code>\@input_(patch)</code> .....           | <u>2552</u>                                                                                                                       |
| <code>\@mainaux</code> .....                 | 774, 887, 892                                                                                                                     |
| <code>\@nobreakfalse</code> .....            | 993                                                                                                                               |
| <code>\@onelevel@sanitize</code> .....       | 2168                                                                                                                              |
| <code>\@onlypreamble</code> .....            | 51, 2473                                                                                                                          |
| <code>\@pkgextension</code> .....            | <u>2525</u>                                                                                                                       |
| <code>\@tfor</code> .....                    | 190, 2700, 2739, 2753,<br>2766, 2783, 2849, 2868, 2883, 2905, 2923, 2942,<br>2962, 2980, 3000, 3017, 3040, 3059, 3077, 3100, 3124 |
| <code>\@whilenum</code> .....                | 2016, 2025                                                                                                                        |
| <code>\bookmark_(patch)</code> .....         | <u>773</u>                                                                                                                        |
| <code>\~</code> .....                        | 1997                                                                                                                              |
| A                                            |                                                                                                                                   |
| <code>\active</code> .....                   | 1970, 2001                                                                                                                        |
| <code>\addcontentsline_(patch)</code> .....  | <u>688</u>                                                                                                                        |
| <code>\AddLineBeginMainAux</code> .....      | 419                                                                                                                               |
| <code>\addtocontents</code> .....            | 1779, 1810, 1815                                                                                                                  |
| <code>\adl@arrayrestore</code> .....         | 75, 82                                                                                                                            |
| <code>\adl@endarray</code> .....             | 75, 82                                                                                                                            |
| <code>\AfterFile</code> .....                | 426, 1749, <u>2536</u>                                                                                                            |
| <code>\aftergroup</code> .....               | 842, 883, 1960                                                                                                                    |
| <code>\AfterPackage</code> .....             | 54,<br>67, 71, 88, 245, 286, 684, 939, 2281, 2500, 2516                                                                           |
| <code>\allpgfmodules</code> .....            | <u>297</u> , 385, 402                                                                                                             |
| <code>\alltikzlibraries</code> .....         | 307, 386                                                                                                                          |
| <code>\alltikzpackages</code> .....          | <u>292</u> , 381, 384, 399                                                                                                        |
| <code>\appendices</code> .....               | <u>2431</u>                                                                                                                       |
| <code>\apptocmd</code> .....                 | 97, 117, 1924                                                                                                                     |
| B                                            |                                                                                                                                   |
| <code>\BeforeFile</code> .....               | 422, 1745, <u>2536</u>                                                                                                            |
| <code>\BeforePackage</code> .....            | 37, 2510                                                                                                                          |
| <code>\BKM@@ref</code> .....                 | 930                                                                                                                               |
| <code>\BKM@color</code> .....                | 805, 815, 823, 835                                                                                                                |
| <code>\BKM@dest</code> .....                 | 926, 928                                                                                                                          |
| <code>\BKM@page</code> .....                 | 910                                                                                                                               |
| <code>\bookmark</code> .....                 | 6, 773, 776, 779, 897, 914, 1945                                                                                                  |
| <code>\BookmarkAtEnd</code> .....            | 893                                                                                                                               |
| <code>\bookmarkget</code> .....              | 851, 853, 861, 863                                                                                                                |
| <code>\bookmarksetup</code> ...              | 6, 719, 721, 723, 811, 829, 856, 866                                                                                              |
| <code>\botmarks</code> .....                 | 1583, 1602, 1604                                                                                                                  |
| <code>\bottomtitles</code> .....             | 5, <u>729</u>                                                                                                                     |
| <code>\bottomtitlespace</code> .....         | 739                                                                                                                               |
| <code>\BS@BKM@FontStyle</code> .....         | 807, 819, 826, 836, 849, 859                                                                                                      |
| C                                            |                                                                                                                                   |
| <code>\C</code> .....                        | 186, 2027, 2034, 2109, 2230                                                                                                       |
| <code>\c</code> .....                        | 2027, 2029, 2233                                                                                                                  |
| <code>\c@lofdepth</code> .....               | 1809, 1810                                                                                                                        |
| <code>\c@lotdepth</code> .....               | 1814, 1815                                                                                                                        |
| <code>\c@tocdepth</code> .....               | 1778, 1779                                                                                                                        |
| <code>\cftdot</code> .....                   | 1661                                                                                                                              |
| <code>\cftdotfill</code> .....               | 1867                                                                                                                              |
| <code>\cftdotsep</code> .....                | 1658, 1870                                                                                                                        |
| <code>\cftnodots</code> .....                | 1660, 1875, 1876                                                                                                                  |
| <code>\changeont</code> .....                | 3, <u>242</u> , 270                                                                                                               |
| <code>\chapter</code> .....                  | 1019, 1830, 2454                                                                                                                  |
| <code>\color</code> .....                    | 231, 232, 273, 277,<br>1089, 1109, 1225, 1237, 1249, 1261, 1273, 1285,<br>1297, 1309, 1321, 1333, 1345, 1357, 2795, 2819, 2821    |
| <code>\colorlet</code> .....                 | 228                                                                                                                               |
| <code>\column</code> .....                   | 2027, 2109, 2140                                                                                                                  |
| <code>\columnsep</code> .....                | 1678, 1707                                                                                                                        |
| <code>\columnseprule</code> .....            | 1677, 1702                                                                                                                        |
| <code>\columnseprulecolor</code> .....       | 1679                                                                                                                              |
| <code>\contentsname</code> .....             | 1736, 1783, 1784, 1904, 1960                                                                                                      |
| <code>\count</code> .....                    | 146                                                                                                                               |
| <code>\countdef</code> .....                 | 2008, 2009, 2010                                                                                                                  |
| <code>\countvalues</code> .....              | 11, <u>1972</u>                                                                                                                   |
| <code>\csepreatpto</code> .....              | 2607, <u>3179</u>                                                                                                                 |
| <code>\csgappto</code> .....                 | 2297, 2535                                                                                                                        |
| <code>\csgpreappto</code> .....              | <u>3179</u>                                                                                                                       |
| <code>\cspreappto</code> .....               | <u>3179</u>                                                                                                                       |
| <code>\csxpreappto</code> .....              | <u>3179</u>                                                                                                                       |
| <code>\CT@cell@color</code> .....            | 2283, 2284                                                                                                                        |
| <code>\CT@cellc</code> .....                 | <u>2281</u>                                                                                                                       |
| <code>\CT@color</code> .....                 | 2283                                                                                                                              |
| <code>\CT@end</code> .....                   | 69, 75                                                                                                                            |
| <code>\currentgrouplevel</code> .....        | 809, 841                                                                                                                          |
| D                                            |                                                                                                                                   |
| <code>\default@color</code> .....            | 232                                                                                                                               |
| <code>\diaghead</code> .....                 | 2276                                                                                                                              |
| <code>\dimen</code> .....                    | 2355, 2356,<br>2357, 2364, 2373, 2375, 2378, 2379, 2384, 2392, 2395                                                               |
| <code>\dimendef</code> .....                 | 2366, 2367, 2368                                                                                                                  |
| <code>\dump</code> .....                     | 114                                                                                                                               |
| E                                            |                                                                                                                                   |
| <code>\embedfile</code> .....                | 2474, 2493                                                                                                                        |
| <code>\embedthisfile</code> .....            | 17, 2469, 2472, 2473                                                                                                              |
| <code>\endarray</code> .....                 | <u>67</u>                                                                                                                         |
| <code>\endlinechar</code> .....              | 424, 428                                                                                                                          |
| <code>\epreatpto</code> .....                | <u>3171</u>                                                                                                                       |
| <code>\eremovefrom</code> .....              | <u>3193</u>                                                                                                                       |
| <code>\eremovefromstart</code> .....         | <u>3217</u>                                                                                                                       |

**F**

[\f@baselineskip](#) ..... 257, 260, 265  
[\f@linespread](#) .. 257, 260, 263, 264, 265, 266, 267, 268  
[\f@size](#) ..... 257, 260, 265  
[\fancy@setoffs](#) .....  
 .. 1456, 1466, 1476, 1486, 1496, 1506, 1516, 1526  
[\fancyhf](#) ..... 1031  
[\fancyhfoffset](#) ..... 1032  
[\figure](#) ..... 1836  
[\firstmarks](#) ..... 1578, 1591, 1601, 1604  
[\fontdimen](#) ..... 174, 180  
[\fontencoding](#) ..... 248  
[\fontfamily](#) ..... 250  
[\fontseries](#) ..... 252  
[\fontshape](#) ..... 254  
[\footrule](#) ..... 1111, 1112, 1113, 1114  
[\footrulewidth](#) ..... 1117, 1118, 1119  
[\FPeval](#) ..... 2162, 2191, 2212, 2213  
[\FPresult](#) ..... 2213, 2214, 2216  
[\futurelet](#) ..... 102, 157, 2806, 2813, 2829, 2837

**G**

[\getrefbykeydefault](#) ..... 910  
[\getthebotmarks](#) ..... [1601](#)  
[\getthefirstmarks](#) ..... [1601](#)  
[\getthemarks](#) .. [10](#), [1591](#), 1601, 1602, 1603, 1607, [1608](#)  
[\getthetopmarks](#) ..... [1601](#)  
[\gluetomu](#) ..... 171  
[\gpreappto](#) ..... [3171](#)  
[\gremovefrom](#) ..... [3193](#)  
[\gremovefromstart](#) ..... [3217](#)

**H**

[\hbs@do\\_\(patch\)](#) ..... [946](#)  
[\hbs@process\\_\(patch\)](#) ..... [956](#)  
[\headrule](#) ..... 1091, 1092, 1093, 1094  
[\headrulewidth](#) ..... 1097, 1098, 1099  
[\headwidth](#) 1098, 1119, 1539, 1540, 1541, 1544, 1545, 1548  
[\htruncadjust](#) ..... [16](#), [2341](#)  
[\htruncwidth](#) ..... [2407](#)  
[\HyperRaiseLinkHook](#) ..... 750, 751, 752, 753  
[\HyperRaiseLinkLength](#) ..... 754

**I**

[\if@nobreak](#) ..... 987  
[\ifcsundef](#) ..... 458,  
 461, 497, 544, 551, 579, 1056, 1060, 1565, 2315  
[\ifcsvoid](#) ..... 637, 638, 639  
[\ifitf@celltranspose](#) ..... 1994, 2038, 2256  
[\ifmarksequal](#) ..... [10](#), [1604](#)  
[\ifnotempty](#) ..... 2512  
[\ifttl@explicit](#) ..... 506  
[\ifundef](#) ..... 1019, 2454, 2501  
[\ignorespaces](#) ..... 2086, 2334, 2354, 2374  
[\in@](#) ... 192, 207, 208, 399, 402, 3214, 3215, 3236, 3238  
[\interitem](#) ..... [15](#), [2293](#)  
[\itf@AfterPackage](#) ..... [64](#)  
[\itf@ampactive](#) ..... 1995, 1998, 2057, 2079  
[\itf@BKM@enable](#) ..... 783,  
 785, 786, 787, 788, 789, 790, 884, 885, 889, 891  
[\itf@BKM@pageref](#) ..... [897](#)  
[\itf@BKM@ref](#) ..... [914](#)  
[\itf@BKM@restore](#) ..... 842, 870, 874, 883  
[\itf@BKM@write](#) ..... 774, [882](#)  
[\itf@cell@calc](#) ..... [2197](#)

[\itf@cell@formulaerror](#) .....  
 .. 2124, 2125, 2127, 2150, 2159, 2162, 2167, 2196  
[\itf@cell@FPcalc](#) ..... 2163, [2197](#)  
[\itf@cell@getformula](#) ..... 2129, [2147](#)  
[\itf@cell@init](#) ..... 2017,  
 2020, 2026, 2027, 2031, 2033, 2037, 2057, 2079  
[\itf@cell@insert](#) ..... 2058, 2080, [2218](#)  
[\itf@cell@numcalc](#) ..... 2140, 2141, 2156, [2197](#)  
[\itf@cell@pgfcalc](#) ..... 2160, [2197](#)  
[\itf@cell@transposefalse](#) ..... 2134  
[\itf@cft@label](#) ..... [1931](#)  
[\itf@cft@newlistof](#) ... [1636](#), 1771, 1772, 1773, 1925  
[\itf@cft@newsection](#) .....  
 ..... 568, [1849](#), 1891, 1894, 1896, 1899, 1901  
[\itf@cft@secname](#) ..... 1825, 1826, 1828, 1884, 1885  
[\itf@cft@sectionname](#) ..... [1825](#)  
[\itf@countvalues](#) ..... 1973, 1975, 1979, 1981  
[\itf@diaghead](#) ..... 2257, 2261, [2273](#)  
[\itf@expandonceifcs](#) ..... 136, 2117, 2145  
[\itf@getthemarks](#) ..... 1594, 1598, 1600  
[\itf@getvalue](#) ..... [1982](#), 2022, 2035  
[\itf@hbs@chkclass](#) ..... 980, 985, 988  
[\itf@hbs@n@breakfix](#) ..... 989, 990  
[\itf@hbs@next](#) ..... 957, 966  
[\itf@hbs@nobreakfix](#) ..... 987, 989, 995  
[\itf@htruncadjust](#) ..... 2342, 2344, 2402  
[\itf@htruncwidth](#) ..... 2408, 2410, 2415, 2417  
[\itf@last@from@next](#) ..... 601, 983, 996  
[\itf@makecell](#) ..... 2006, 2007, 2269  
[\itf@marks@new](#) ..... 1557, 1563, [1571](#)  
[\itf@message](#) ..... 2083, [2088](#)  
[\itf@musk](#) ..... 128, 1657, 1658, 1872, 1873  
[\itf@number](#) ..... [145](#), 2095, 2114, 2142, 2305  
[\itf@pagesetup](#) ..... 1036, 1037, 1039, 1041  
[\itf@patch](#) [102](#), 691, 773, 946, 956, 1903, 1910, 1917, 1924  
[\itf@pgfk@color](#) ..... 2794, 2802, 2806, 2813  
[\itf@pgfk@colorcmd](#) ..... 2817, 2825, 2829, 2837  
[\itf@pgfkeys@aliaserror](#) ..... 2688, 2690, 2696  
[\itf@pgfkeys@case@three](#) 2566, 2572, 2662, 2671, 2673  
[\itf@pgfkeys@case@three@handleall](#) .. 2567, 2673  
[\itf@pgfkeys@default@blank@required](#) .....  
 ..... 2622, 2632, 2637, 2666  
[\itf@pgfkeys@ifaliasexists](#) 2677, 2684, 2689, 2698  
[\itf@preappto](#) ..... 3184  
[\itf@preapptocmd](#) ..... [95](#), 105, 691  
[\itf@rem@mu](#) ..... 164, 168  
[\itf@removefrom](#) ..... 3197, [3211](#)  
[\itf@removefromstart](#) ..... 3221, [3235](#)  
[\itf@repeatcell](#) ..... 2002, 2004, 2087  
[\itf@resetlastsections](#) ..... 1004, 1007, 1012  
[\itf@scaledimen](#) ..... 2377, 2392, 2404, 2406  
[\itf@scalefactor](#) ..... 2403, 2405  
[\itf@sectionformat](#) ..... 436, 437, 439  
[\itf@setbottomtitlespace](#) ..... 732, 735, 738  
[\itf@setlist](#) ..... 2297, 2298, 2300, 2321  
[\itf@setlist@error](#) ..... 2316, 2322, 2327  
[\itf@sk](#) ..... 127, 733, 736, 739,  
 3097, 3098, 3103, 3105, 3107, 3110, 3112, 3127, 3133  
[\itf@strip@at](#) ..... [131](#), 484, 947  
[\itf@strip@dot](#) ..... [131](#), 173, 179  
[\itf@strip@mu](#) ..... [163](#), 1658, 1873  
[\itf@tk](#) ..... 129, 1537  
[\itf@tocloft](#) ..... 1761, 1904, 1911, 1918, 1931, 1935  
[\itf@ttl@BKM](#) ..... 692, 702, 728  
[\itf@ttl@createsection](#) ..... [557](#), 580

|                                               |                                                                    |                                                                       |                                           |
|-----------------------------------------------|--------------------------------------------------------------------|-----------------------------------------------------------------------|-------------------------------------------|
| <code>\itf@ttl@getformat</code> .....         | 449                                                                | <code>\pagestyle</code> .....                                         | 1048, 1056, 1079, 1546                    |
| <code>\itf@ttl@getspacing</code> .....        | 449                                                                | <code>\paragraph</code> .....                                         | 1834                                      |
| <code>\itf@ttl@newsection</code> .....        | 457, 576                                                           | <code>\patchcmd</code> .....                                          | 104, 126, 773                             |
| <code>\itf@ttl@sectionformat</code> .....     | 448                                                                | <code>\pdfstringdefDisableCommands</code> .....                       | 246                                       |
| <code>\itf@ttl@setkey</code> .....            | 465,<br>469, 471, 472, 473, 474, 475, 486, 487, 488, 489, 490      | <code>\pgfkeys@case@three</code> .....                                | 2566, 2572, 2661                          |
| <code>\itf@ttl@setup</code> .....             | 449                                                                | <code>\pgfkeys@case@three@handleall</code> .....                      | 2567                                      |
| <code>\itf@ttl@titleclass</code> .....        | 524, 584, 586                                                      | <code>\pgfkeys@exp@call</code> .....                                  | 2574, 2617                                |
| <code>\itf@ttl@titleclass@i</code> .....      | 527, 528, 530, 542                                                 | <code>\pgfkeys@global@temp</code> .....                               | 2592, 2599, 2608, 2615                    |
| <code>\itf@ttl@titleclass@ii</code> .....     | 538, 543, 556                                                      | <code>\pgfkeys@ifexecutehandler</code> .....                          | 2563, 2590, 2602, 2606, 2619, 2623, 2672  |
| <code>\itf@ttl@update</code> .....            | 452, 492                                                           | <code>\pgfkeys@unpack</code> .....                                    | 2565, 2571                                |
| <code>\itf@undefinedpagestyle</code> ...      | 1057, 1061, 1064, 1067                                             | <code>\pgfkeysaddvalue_(bug_correction)</code> .....                  | 2590                                      |
| <code>\itf@units@mu</code> .....              | 183, 1657, 1872                                                    | <code>\pgfkeysEaddvalue</code> .....                                  | 1731, 1737, 2130, 2602                    |
| <code>\itf@units@mutopt</code> .....          | 170, 211                                                           | <code>\pgfkeysEsetvalue</code> .....                                  | 2602                                      |
| <code>\itf@units@normalize</code> .....       | 224                                                                | <code>\pgfkeysmeaning</code> .....                                    | 2581                                      |
| <code>\itf@units@pt</code> .....              | 202                                                                | <code>\pgfkeysnovalue</code> 1647, 2647, 2682, 2791, 2799, 2807, 2814 |                                           |
| <code>\itf@units@pttomu</code> .....          | 170, 196                                                           | <code>\pgfkeysvalueof@unexpanded</code> .....                         | 2586, 2742, 2756, 2852, 2945, 2983        |
| <code>\itf@units@scale</code> .....           | 216, 2947,<br>2966, 2985, 3004, 3026, 3046, 3064, 3085, 3107, 3132 | <code>\pgfkeysvaluerequired</code> .....                              | 2652                                      |
| <code>\itf@units@scale@normalize</code> ..... | 218, 219                                                           | <code>\pgfmthparse</code> .....                                       | 2159, 2187, 2207                          |
| <code>\itf@usetikzcode</code> .....           | 388, 397                                                           | <code>\pgfmthresult</code> .....                                      | 2208                                      |
| <b>L</b>                                      |                                                                    | <code>\preappto</code> .....                                          | 3171                                      |
| <code>\label</code> .....                     | 1956                                                               | <code>\pretocmd</code> ..                                             | 96, 117, 946, 956, 1760, 1903, 1910, 1917 |
| <code>\lastchapterbookmark</code> .....       | 1022                                                               | <b>R</b>                                                              |                                           |
| <code>\lastchaptertitle</code> .....          | 1020                                                               | <code>\R</code> .....                                                 | 2017, 2021, 2225                          |
| <code>\lastchaptertoc</code> .....            | 1021                                                               | <code>\r</code> .....                                                 | 2017, 2019, 2228                          |
| <code>\lastsection</code> .....               | 954, 964                                                           | <code>\raisedhyperdef</code> .....                                    | 6, 746, 1941                              |
| <code>\lastsectionname</code> .....           | 991, 992, 997, 1013                                                | <code>\rem@pt</code> .....                                            | 264                                       |
| <code>\listfigurename</code> .....            | 1795, 1796, 1911                                                   | <code>\remove@to@nnil</code> .....                                    | 141, 1977, 1987, 1989, 2165               |
| <code>\listofsetup</code> .....               | 9, 1624, 1768, 1930                                                | <code>\removefrom</code> .....                                        | 22, 3193                                  |
| <code>\listtablename</code> .....             | 1801, 1802, 1918                                                   | <code>\removefromstart</code> .....                                   | 2090, 2092, 3217                          |
| <code>\LoadInterface</code> .....             | 6, 7, 9, 39                                                        | <code>\ReNewEnviron</code> .....                                      | 2501, 2502                                |
| <code>\lofsetup</code> .....                  | 1624, 1916                                                         | <code>\RenewEnviron</code> .....                                      | 2502                                      |
| <code>\lotsetup</code> .....                  | 1624, 1923                                                         | <code>\repeatcell</code> .....                                        | 11, 1994, 2146, 2178, 2181                |
| <code>\ltx@clsextension</code> .....          | 2527                                                               | <code>\row</code> .....                                               | 2017, 2141                                |
| <code>\ltx@ifclassloaded</code> .....         | 2529                                                               | <b>S</b>                                                              |                                           |
| <code>\ltx@ifpackageloaded</code> .....       | 52, 56, 65, 72, 89, 226, 230, 1668, 2528                           | <code>\scr@load@hook</code> .....                                     | 2531, 2553, 2555                          |
| <code>\ltx@pkgextension</code> .....          | 2526                                                               | <code>\scr@set@load@hook_(optimisation)</code> .....                  | 2535                                      |
| <b>M</b>                                      |                                                                    | <code>\sectionformat</code> .....                                     | 4, 435, 721                               |
| <code>\makecell</code> .....                  | 2007                                                               | <code>\selectfont</code> .....                                        | 243                                       |
| <code>\marks</code> .....                     | 42, 1559, 1567                                                     | <code>\set@fontsize</code> .....                                      | 257, 260, 265                             |
| <code>\marksthe</code> .....                  | 10, 631, 1555                                                      | <code>\setlist</code> .....                                           | 15                                        |
| <code>\marksthecs</code> .....                | 1561                                                               | <code>\setlist_(extension, _star_form)</code> .....                   | 2296                                      |
| <code>\maxdimen</code> .....                  | 2333                                                               | <code>\skip</code> .....                                              | 170, 183, 202                             |
| <code>\muexpr</code> .....                    | 178                                                                | <code>\strip@pt</code> .....                                          | 2403                                      |
| <code>\multicolsep</code> .....               | 1675                                                               | <code>\strut</code> .....                                             | 1075                                      |
| <code>\muskip</code> .....                    | 170, 183, 202                                                      | <code>\subparagraph</code> .....                                      | 1835                                      |
| <code>\mutoglu</code> .....                   | 178                                                                | <code>\subsubfigure</code> .....                                      | 1837                                      |
| <b>N</b>                                      |                                                                    | <code>\subsubsection</code> .....                                     | 1833                                      |
| <code>\newcounter</code> .....                | 545                                                                | <code>\subtable</code> .....                                          | 1839                                      |
| <code>\newlist</code> .....                   | 2325                                                               | <b>T</b>                                                              |                                           |
| <code>\newlistentry_(patch)</code> .....      | 1903                                                               | <code>\tab</code> .....                                               | 1996, 2000                                |
| <code>\newmarks</code> .....                  | 1572                                                               | <code>\table</code> .....                                             | 1838                                      |
| <code>\newmuskip</code> .....                 | 128                                                                | <code>\textfont</code> .....                                          | 174, 180                                  |
| <code>\newskip</code> .....                   | 127                                                                | <code>\textheight</code> .....                                        | 736                                       |
| <code>\newtoks</code> .....                   | 129                                                                | <code>\the@dimen</code> .....                                         | 2348, 2386, 2395                          |
| <code>\normalbaselineskip</code> .....        | 1140                                                               | <code>\thebotmarks</code> .....                                       | 1576                                      |
| <code>\normalcolor</code> .....               | 1712, 2797, 2823, 2832, 2840                                       | <code>\thefirstmarks</code> .....                                     | 10, 1576                                  |
| <b>P</b>                                      |                                                                    | <code>\thepage</code> .....                                           | 906, 923                                  |
| <code>\pagesetup</code> .....                 | 9, 1033, 1077, 1537                                                | <code>\thetopmarks</code> .....                                       | 1576                                      |
|                                               |                                                                    | <code>\thispagestyle</code> .....                                     | 1060                                      |
|                                               |                                                                    | <code>\tikzset</code> .....                                           | 288                                       |

|                                       |                              |                                      |                           |
|---------------------------------------|------------------------------|--------------------------------------|---------------------------|
| <code>\titleclass</code> .....        | 418, 532, 535, 560           | <b>U</b>                             |                           |
| <code>\titleformat</code> .....       | 418                          | <code>\unless</code> .....           | 991, 992                  |
| <code>\titlespacing</code> .....      | 418                          | <code>\unskip</code> .....           | 2334, 2354, 2374          |
| <code>\tocloftpagestyle</code> .....  | 1638                         | <code>\usepgfmodule</code> .....     | 385, 403                  |
| <code>\tocsetup</code> .....          | 8, 8, <u>1624</u> , 1909     | <code>\usetikz</code> .....          | 3, 389, <u>390</u> , 2187 |
| <code>\toks</code> .....              | 2356, 2362, 2377, 2381, 2392 | <code>\usetikzlibrary</code> .....   | 375, 386, 404             |
| <code>\toksdef</code> .....           | 2011                         | <b>V</b>                             |                           |
| <code>\topmarks</code> .....          | 1588, 1603                   | <code>\vsize</code> .....            | 2333                      |
| <code>\truncate</code> .....          | 2336                         | <code>\vskip</code> .....            | 1099, 1118, 1119          |
| <code>\TruncateMarker</code> .....    | 2331                         | <b>W</b>                             |                           |
| <code>\truncatext</code> .....        | 16, <u>2331</u>              | <code>\write</code> .....            | 774, 887, 892             |
| <code>\ttl@getkeys</code> .....       | 494, 495                     | <b>X</b>                             |                           |
| <code>\ttl@spacing@ii</code> .....    | 517                          | <code>\xpreappto</code> .....        | <u>3171</u>               |
| <code>\ttlh@runinifempty</code> ..... | <u>430</u>                   | <code>\xremovefrom</code> .....      | <u>3193</u>               |
| <code>\ttlp@append</code> .....       | 414, 420                     | <code>\xremovefromstart</code> ..... | <u>3217</u>               |