The LaTeX keyfloat Package

v0.15 — 2017/05/12

bd@BDTechConcepts.com

Provides a key/value interface for generating floats.

Abstract

The keyfloat package provides a key/value user interface for quickly creating figures with a single image each, figures with arbitrary contents, tables, subfloats, rows of floats, floats located [H]ere, floats in the [M]argin, and floats with text [W]rapped around them.

Key/value combinations may specify a caption and label, a width proportional to `\linewidth`, a fixed width and/or height, rotation, scaling, a tight or loose frame, an `\arraystretch`, a continued float, additional supplemental text, and an artist/author's name with automatic index entry. When used with the tocdata package, the name also appears in the List of Figures.

Floats may be placed into a multi-row environment, and are typeset to fit within the given number of columns, continuing to the next rows as necessary. Nested sub-rows may be used to generate layouts such as two small figures placed vertically next to one larger figure.

Subfloats are supported by two environments.

As an example, a typical command to include a figure with a framed image of half `\linewidth` could be:

```
\keyfig*[hbp]{f,lw=.5,c={A caption},l={fig:label}}{image}
```

Contents

List of Examples

List of Figures

List of Tables

1 Introduction

The keyfloat package simplifies the creation of LaTeX floats, while still allowing a number of useful features.

1.1 A Problem with Floats

When including a figure with a graphics image into a document, the user typically enters something such as:

```
\begin{figure}
\centering
\includegraphics[width=3in]{filename}
\caption{A Figure}
\label{fig:somelabel}
\end{figure}
```

When doing that often enough, it makes sense to factor the common code:

```
\onefigure[3in]{filename}{A Figure}{fig:somelabel}
```

Expanding the capability of `\onefigure` via xparse can lead to the general case of:

```
\onefigure*[loc](width){filename}(add'l text)[shortcap]{caption}*[label]
```

Attempting to add additional features such as frames and continued floats hits the limit of nine parameters for a TeX macro, requiring that new features use some kind of change-state macros instead. Attempting to support rows of floats or subfloats only makes things more complicated still.

A key/value system solves the problem of adding more features, does not require much additional typing, is a more self-documenting syntax, and allows a shared syntax with subfloats and groups of floats as well. Thus, the keyfloat package.

1.2 The keyfloat Package

Using keyfloat, the previous example becomes:

```
\keyfig{w=3in,c=A Figure,l=fig:somelabel}{filename}
```

The `\onefigure` general case becomes:

```
\keyfig*[loc]{w=width,t={add'l text},sc=shortcap,cstar=caption,
    l=label}{filename}
```

1.3 Features

The macros and environments provided by keyfloat include:

`\keyfig`: A figure with an image.

`\keyfigbox`: A figure with arbitrary contents.

`\keyparbox`: A "figure" without a caption, useful to place uncaptioned text inside a group,

`\keytab`: A table.

keyfigure: A figure environment.

keytable: A table environment.

keyfloats: A group of rows and columns of floats.

keysubfigs: A figure containing a group of rows and columns of subfigures.

keysubtabs: A table containing a group of rows and columns of subtables.

keywrap: Wraps a keyfloat around an environment of text. Usable inside a list.

marginfigure: A figure environment placed into the margin.[1]

margintable: A table environment placed in the margin.

Additional features include:

- Rows and columns of floats may be generated by placing them inside a `keyfloats` environment.
- Subfloats may be generated by placing them inside a `keysubfigs` or `keysubtabs` environment.
- Dynamic layout: The number of columns is specified. Extra floats are placed onto additional rows as needed, with the final row adjusted to compensate for leftovers.

[1] `marginfigure` and `margintable`: The environments provided by the `tufte-book` class are used if loaded, otherwise keyfloat provides its own versions.

- Floats may be placed [H]ere.
- Floats may be placed in the [M]argin.
- Floats may be placed with text [W]rapped around them.
- Floats may be starred to span two columns.
- Continued floats may be used to repeat the previous float number.
- A figure may contain an image, with additional sizing, rotation, and a frame.
- Tables may be stretched.
- Boxes of arbitrary contents may be assigned a width and framed.
- Floats may be moved into and out of the grouping environments as needed.
- An artist/author's name may be added to a figure and the index.
- If the tocdata package is loaded (use v0.12+), the name is also added to the LOF.
- Additional descriptive text may be added as well.
- Frames may be customized.

examples — A large number of examples are provided, each showing LaTeX source and the resulting float.

index — A customized index is included at the back of the documentation.

margin tags — Blue margin tags are used to help quickly find information, and often indicate the destination of index entries.

⚠ warnings — Several warnings are noted in the text. Watch out for these special cases.

problems — See the "troubleshooting" section of the index for help with specific problems which may occur.

2 Using the keyfloat Package

2.1 Loading keyfloat

keyfloat is loaded with the usual command:

```
\usepackage{keyfloat}
```

If you wish to have artist's names appear in the table of contents, as provided by the tocdata package, load either tocloft or titletoc, followed by tocdata, then keyfloat:

```
\usepackage{titletoc}% or titletoc
\usepackage{tocdata}
\usepackage{keyfloat}
```

2.2 Macros and Environments

`\keyfig` * [⟨*loc*⟩] {⟨*keys/values*⟩} {⟨*image filename*⟩}
A macro to generate a figure with an image from a file.

`\keyfigbox` * [⟨*loc*⟩] {⟨*keys/values*⟩} {⟨*box contents*⟩}
A macro to generate a figure with arbitrary paragraph contents. See example 2.

`\keyparbox` * [⟨*loc*⟩] {⟨*keys/values*⟩} {⟨*box contents*⟩}
A macro to generate a figure with arbitrary paragraph contents, but no number or caption. This is equal to a `\keyfigbox` with `cstar={}`. Mostly useful to add supplemental information inside a row of floats or subfloats. See example 14.

`\keytab` * [⟨*loc*⟩] {⟨*keys/values*⟩} {⟨*tabular contents*⟩}
A macro to generate a table with tabular contents. Usually use the `keytable` environment instead.

Env `keyfigure` * [⟨*loc*⟩] {⟨*keys/values*⟩}
An environment to generate a figure with arbitrary contents. Useful for multi-paragraph contents. See example 3.

Env `keytable` * [⟨*loc*⟩] {⟨*keys/values*⟩}
An environment to generate a table with arbitrary contents. Useful for larger tables. See example 5.

The above macros and environments may be used by themselves, or inside the following `keyfloats`, `keysubfigs`, or `keysubtabs` environments.

Env **keyfloats** * [⟨*loc*⟩] {⟨*num columns*⟩}
A group of figures or tables typeset in rows. May be nested. See example 15.

Env **keysubfigs** * [⟨*loc*⟩] {⟨*numcols*⟩} {⟨*keys*⟩}
A group of subfigures typeset in rows. May not be nested. See example 16.

Env **keysubtabs** * [⟨*loc*⟩] {⟨*numcols*⟩} {⟨*keys*⟩}
A group of subtables typeset in rows. May not be nested. See example 17.

Env **keywrap** {⟨*width of keyfloat*⟩} {⟨*keyfloat*⟩}
Displays a keyfloat next to an environment of text. Usable inside a list item, where [W] will not work. ⟨*keyfloat*⟩ may be any of **\keyfig**, **keyfigure**, **keyfloats**, **keysubfigs**, etc., each with its proper arguments. See example 27.

Env **marginfigure** [⟨*offset*⟩]
A figure placed into the margin, with an optional vertical offset. **\keyfloat** uses the version provided by the **tufte-book** class if available, or provides its own version otherwise. See example 20.

Env **margintable** [⟨*offset*⟩]
A table placed into the margin, with an optional vertical offset. **\keyfloat** uses the version provided by the **tufte-book** class if available, or provides its own version otherwise. See example 21.

Arg * The star option create floats which span both columns in a two-column document.

Arg [H] The [H] location forces a figure to be "Here", in the form of a minipage instead of a float. A caption, label, etc. may still be assigned.

Arg [M] The [M] location places the float into the margin. When the **tufte-book** class is used, its **marginfigure** and **margintable** environments are used, otherwise keyfloat provides and uses its own versions of the same environments. See examples 22 and 23.

Arg [W] The [W] location wraps text around the float. Use this just before the start of a paragraph with contents large enough to wrap around the float. Do not use
Pkg **wrapfig** this inside a list environment. Select placement with the **wp** key; see the wrapfig package documentation for more information.

Arg [loc] The star and [loc] options are ignored for floats inside a **keyfloats**, **keysubfigs**, or **keysubtabs** environment. Note that these container environments may have their own star and [loc] options.

2.3 Keys and Values

Table 1 shows the key/value combinations which are allowed. In most cases these may be used in any order and any combination, except for the following:

subfloat keys

The keys labeled "Sub" may be used for the `keysubfigs` and `keysubtabs` environments, which group a number of subfloats together under one master float. The master float has its own caption, label, and text, and each subfloat inside the group likewise has its own set of keys.

keyfloats keys

`keyfloats` does not accept any keys at all.

The "artist" keys `ap`, `af`, `al`, and `as` are only used by figures.

The `stretch` key increases space between tabular elements.

The rest of the macros and environments accept all of the keys, as they each create an individual float or subfloat, and each may have its own assigned dimensions and frame.

short/long caption combinations

Table 2 shows the combinations of the caption-related keys `c`, `cstar`, and `sc`, and how they control the caption numbering and entries in the LOF/LOT.

wrapped float placement

Table 3 shows the wrapped-float placement options for the `wp` key for floats placed `[W]`.

Table 1: Keys and Values — Part I

Key	Sub[a]	Description	Example
`c`	•	An unstarred caption. If empty, creates a figure with a number but no caption.	`c=A Caption`
`cstar`	•	A starred caption. Creates a float without a number. If empty, creates a figure with no number or caption.	`cstar=No Num`
`sc`	•	The short caption for the LOF/LOT, even if cstar.	`sc=Short Cap`
`cont`	•	Continued float?	`cont`
`l`	•	The label. Enclose in braces if a comma is included. Ignored in unnumbered floats.	`l=fig:A name`
`ap`	•	Artist's prefix, such as "Mr."[b]	`ap=Mr.`
`af`	•	Artist's first name.[b]	`af=First`
`al`	•	Artist's last name.[b]	`al=Last`
`as`	•	Artist's suffix, such as `~III`.[b]	`al=~III`
`t`	•	Additional text. May include paragraphs. Enclose in braces if a comma is included. May need `\protect` before macro calls. Fully-justified alignment.	`t=Paragraphs`
`tc`	•	Additional text, aligned to the center.	`tc=Paragraphs`
`tl`	•	Additional text, aligned to the left.	`tl=Paragraphs`
`tr`	•	Additional text, aligned to the right.	`tr=Paragraphs`

[a:] All the keys in Part I may be used with the `keysubfigs` and `keysubtabs` environments.

[b:] Artist keys: Only used in Figure floats. A fixed-width non-breakable space is placed between names, except that the optional suffix is connected directly to the last name, allowing "`as={, Title}`", for example.

... continued

Table 1: Keys and Values — Part II

Key[a]	Description	Example
`lw`	Set the width to a fraction of `\linewidth`. Cancels `w`. If a non-image float, sets the width of the text box.	`lw=.5`
`w`	Set the actual width. Cancels `lw`. If a non-image float, sets the width of the text box.	`w=2in`
`h`	Set the actual height, images only.	`w=2in`
`s`	Set the image scale, images only.	`s=3`
`a`	Set the rotation angle; counter-clockwise degrees.	`r=90`
`f`	Selects a loose frame with the current `\fboxsep`. Only rotated with `\keyfig`.	`f`
`ft`	Selects a tight frame with no `\fboxsep`. Useful for photographs, or diagrams which already have some margin built in.	`ft`
`stretch`	Sets `\arraystretch` inside the float.	`stretch=1.5`
`mo`	Sets the vertical offset for a margin float.	`mo=-1.2ex`
`wp`	Sets the wrap placement for a wrapped float. The default is `O`, which places the wrapped float at the outside edge of the text. See table 3.	`wp=I`
`va`	Sets the vertical alignment of the outermost minipage container for the keyfloat. Defaults to 'c'.	`va=t`

[a:] None of the keys in Part II are used in the `keysubfigs` and `keysubtabs` environments.

Table 2: Caption-Related Key Combinations

Keys in Use			Type of	
`c`	`cstar`	`sc`	**Caption**[a]	**LOF/LOT**[b]
•	—	—	Numbered	Caption
•	—	•	Numbered	Short Caption
—	•	—	Unnumbered	None
—	•	•	Unnumbered	Short Caption
—	`cstar={}`	Ignored	None	None

[a:] Caption: Shows whether the float will be numbered, unnumbered, or have no caption.

[b:] LOF/LOT: Shows whether the regular or short caption will appear in the List of Figures or List of Tables, or if there will be no listing.

Table 3: Wrapped-Float Placement Options

Key		Location
r	R	to the right of the text
l	L	to the left of the text
i	I	to the inside margin
o	O	to the outside margin

The un-capitalized key attempts to place the float "here", and the capitalized key allows LaTeX to try to find the best location. The default is `O`.

2.4 Other Settings

`\KFLTtightframe` `{`⟨*contents*⟩`}` Frames the contents without separation.

`\KFLTlooseframe` `{`⟨*contents*⟩`}` Frames the contents with separation.

These may be used to re-define how contents are framed. The default is a simple `\fbox`.

Len `\KFLTtightframewidth` Combined width of the frame and separation for each of tight and loose frames. These settings should be adjusted when changing the frame width and/or separation.

Len `\KFLTlooseframewidth` The value should be equivalent to `\fboxwidth` plus `\fboxsep`.

Len `\KFLTimageboxwidth` The computed width of the image. Useful to enclose an mdframed environment to restrict its width. See example 28.

Figure 1: A `\keyfig` with an image

Some text. More text.

Another paragraph.

Figure 2: A `\keyfigbox`

2.5 Examples

2.5.1 Single Floats

Example 1: Figure with an image from a file

Code:

```
\keyfig{c=A \cs{keyfig} with an image,l=fig:simple}{image}
```

Result:
Figure 1

natural size

This float (fig. 1) is shown at its natural size because no width or height modifiers were specified. When used alone like this, a regular float is created.

Example 2: Figure with arbitrary contents

Code:

```
\keyfigbox{f,c={A \cs{keyfigbox}},l=fig:figbox}
    {Some text.  More text. \par  Another paragraph.}
```

Result:
Figure 2

default width

The `\keyfigbox` creates a figure with a box of arbitrary contents, instead of an image from a file. Its default width is the full `\linewidth`, unless `w` or `lw` keys are used.

Arbitrary contents may go here.

Including multiple paragraphs.

Figure 3: A `keyfigure` environment

Table 4: A `\keytab` table

A	B
C	D

Example 3: Figure environment with arbitrary contents

Code:

```
\begin{keyfigure}{f,c={A \env{keyfigure} environment},
    l=fig:environment}
Arbitrary contents may go here.

Including multiple paragraphs.
\end{keyfigure}
```

Result:

Figure 3

The `keyfigure` environment is preferred over the `\keyfigbox` macro when multiple lines of contents are to be included.

Example 4: Table macro

Code:

```
\keytab{c=A \cs{keytab} table,l=tab:simpletable}{\testtable}
```

Result:

Table 4

Do not try to use tables which overflow the page.

For anything other than a simple table, use the `keytable` environment. See example 5.

large tables For large tables, use the longtable or supertabular packages.

Table 5: A `keytable` environment

Arbitrary contents may go here.[a]

A	B
C	D

[a] A footnote.

Example 5: Table environment with arbitrary contents

Code:

```
\begin{keytable}{f,c={A \env{keytable} environment},
    l=tab:environment}
Arbitrary contents may go here.\footnote{A footnote.}

\testtable
\end{keytable}
```

Result:

Table 5

The `keytable` environment is preferred over the `\keytab` macro since most tables are multi-line creations.

Additional text. Multiple paragraphs may be used. The entire text is enclosed in braces because a comma is included. Alignment may be set by using tags `tc`, `tl`, or `tr` instead of `t`

Figure 4: A figure with many options

Example 6: Figure with many options selected

Code:

```
\keyfig{
    w=2in,ft,r=15,
    c=A figure with many options,
    sc=A figure with options,
    t={Additional text.  Multiple paragraphs may be used.
    The entire text is enclosed in braces because a comma
    is included. Alignment may be set by using
    tags \texttt{tc}, \texttt{tl}, or \texttt{tr}
    instead of \texttt{t}},
    l=fig:options
}{image}
```

Result:

Figure 4

Width is fixed at 2 in, a tight frame is specified (`\fboxsep` of 0 pt), a short caption appears in the List of Figures, and the additional text is using the default fully-justified alignment.

Since fig. 4 is a float, it may appear on the following page.

Figure 5: Half of `\linewidth`

Example 7: Using `\linewidth`

Code:

```
\keyfig{lw=.5,c=Half of \cs{linewidth},l=fig:linewidth}{image}
```

Result:

Figure 5

`\linewidth` Figure 5 is half of `\linewidth` in size. When the `lw` key is used inside a `keyfloats` or `keysubfigs` environment, the `\linewidth` will be proportional to the sub-box for each element. When used alone, such as here, the `\linewidth` is the full width of the text on this page.

`lw` and `w` are not used at the same time. If both `lw` and `w` are specified, the last one cancels any previous ones.

Example 8: Using frames

Code:

```
\begin{keyfloats}[hbp]{4}
\keyfig{f,c=Loosely-framed figure,l=fig:looseframe}{image}
\keyfig{ft,c=Tightly-framed figure,l=fig:tightframe}{image}
\keytab{f,c=Loosely-framed table,l=tab:looseframe}{\testtable}
\keytab{ft,c=Tightly-framed table,l=tab:tightframe}{\testtable}
\end{keyfloats}
```

Result:

Figures 6 and 7 and tables 6 and 7

Figure 6: Loosely-framed figure

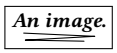


Figure 7: Tightly-framed figure

Table 6: Loosely-framed table

A	B
C	D

Table 7: Tightly-framed table

A	B
C	D

The `f` key adds a loose frame with the current `\fboxsep`. This is desirable in most cases.

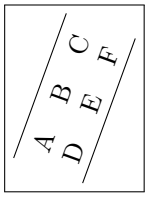
The `tf` key adds a tight frame with no separation. This is useful for framing a photograph, or a diagram which already has a margin.

Framing tables is seldom recommended. In the case of the tight frame, table 7, note that the external frame almost overwrites the table's natural horizontal rules.

custom frames

Also see section 2.6.1 for customizing frames.

Table 8: Table, rotated



(Framed to show box width.)

Example 9: Using rotation with boxes

Code:

```
\keytab{f,w=.8in,c={Table, rotated},
    r=70,l=tab:rotated,
    tc=(Framed to show box width.)}
    {\testwidetable}
```

Result:

Table 8

rotated whitespace

⚠ box width

Unless a width is given, a box is the full `\linewidth`. When rotated, this extra horizontal space is rotated into extra vertical space. To avoid this extra space, set a `w` or `lw` to be wide enough for the table or other contents, but not much wider. When this box is rotated, it will not take much more vertical space than necessary.

frame rotation

Unlike an image, the frame of a box does not rotate with its contents.

Example 10: Located [H]ere

Code:

```
\keytab[H]{c={A table [H]},l=fig:here}{\testtable}
\keyfig[H]{f,w=1in,c={A keyfig [H]}}{image}
```

Result:

Table 9

Table 9: A table [H]

A	B
C	D

Figure 8: A keyfig [H]

⚠ Out of sequence

Table 9 is to be placed "[H]ere", and therefore may appear out-of-sequence with surrounding figures. Place a `\clearpage` before or after to re-sync, if necessary.

Starred caption with a short caption.

Example 11: Unnumbered float

Code:

```
\keyfig[H]{f,cstar={A starred caption}}{image}
```

Result:

See fig: "A starred caption".

A starred caption

A starred caption creates a float without a number, and without an entry in the List of Figures unless there is a non-empty short caption. (See the next example.)

⚠ No label Labels cannot be used when there is no number for a float.

Example 12: Unnumbered float with a lof entry

Code:

```
\keyfig{
    f,cstar={Starred caption with a short caption.},
    sc={Starred short caption}
}{image}
```

Result:

See fig: "Starred caption with a short caption".

A starred caption with a non-empty short caption creates an unnumbered entry in the List of Figures.

Example 13: An unnumbered in-text image

Code:

```
\keyfig[H]{f,cstar={},
    tc={Optional text which is not a caption.}
}{image2}
```

Result:

See fig: "Optional text which is not a caption."

Optional text which is not a caption.

By using [H] and `cstar={}`, the image is placed inline without a number or LOF entry.

Also see example 14.

Some contents.

A \keyparbox with no number or label.

Figure 9: Next to a \keyparbox

Example 14: A box without a caption.

Code:

```
\begin{keyfloats}{2}
\keyparbox{
    f,lw=.5,
    tc={A \cs{keyparbox} with no number or label.}
}{Some contents.}
\keyfig{c=Next to a \cs{keyparbox},l=fig:nexttoparbox}{image}
\end{keyfloats}
\keyparbox[H]{f,lw=.5}{A \cs{keyparbox} [H], outside the row.}
```

Result:

Figure 9, and the box to its left.

A \keyparbox [H], outside the row.

A \keyparbox is a \keyfigbox with cstar={}, and is mostly useful as an information box inside a row or a set of subfloats.

2.5.2 Groups of Floats

Example 15: Groups of figures — `keyfloats` environment

Code:

```
\begin{keyfloats}{2}
\keyfig{lw=1,f,c={First in a group},
    l=fig:firstinrow,tl={\cs{raggedright} text}
    }{image}
\keyparbox{}{\centering A \cs{keyparbox} describing something.
    \par With several paragraphs.}
\begin{keyfloats}{2}
\keyfig{lw=1,c={Third in a group},
    l=fig:thirdinarow}{image}
\keyfig{lw=1,c={Fourth in a group}}{image2}
\keyfig{lw=1,c={Fifth in a group}}{image}
\keyfig{lw=1,c={Sixth in a group},
    l=fig:sixthinarow}{image2}
\end{keyfloats}
\keytab{c={Seventh in a group},l=tab:seventhinrow}{\testwidetable}
\end{keyfloats}
```

Result:

Figure 10 to Table 10

Figure 10 to table 10 are in a `keyfloats` environment. Furthermore, Figures 11 to 14 are in an additional nested `keyfloats` environment, forming a small box of floats inside the larger group.

The `keyfloats` environment takes an argument for the number of columns. Additional floats are automatically placed on following rows. Changing the number of columns will cause the floats to automatically readjust as necessary. Leftovers will be centered on the last row.

⚠ `\linewidth` Note that `\linewidth` is adjusted for each row and nested row, so the `lw` key will need to be changed if a float is moved to a different nesting level.

⚠ image too large Fixed-width or fixed-height floats may be too large to fit if they are moved into a group. It is the user's responsibility to adjust `w`, `h`, or `lw` as necessary.

Keyfloats may be positioned [H]:
`\begin{keyfloats}[H]{2}` ...

Keyfloats may be starred to span both columns in a two-column format:
`\begin{keyfloats}*{2}` ...

\raggedright text

Figure 10: First in a group

Figure 11: Third in a group

Figure 12: Fourth in a group

Figure 13: Fifth in a group

Figure 14: Sixth in a group

A \keyparbox describing something.

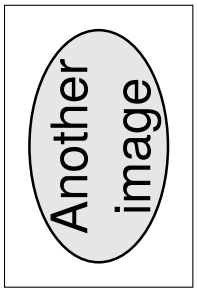
With several paragraphs.

Table 10: Seventh in a group

A	B	C
D	E	F

Some text

a: First subfigure



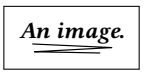
Lots of lots of lots of lots of text.

b: Second subfigure

c: Third subfigure

d: Fourth subfigure

A	B
C	D



e: Fifth subfigure

Figure 15: Subfigures

2.5.3 Subfloats

Example 16: Subfigures — `keysubfigs` environment

Code:

```
\begin{keysubfigs}{3}{c=Subfigures,l=fig:subfigs}
\keyfig{lw=1,f,c={First subfigure},
    l=fig:firstsubfig,t=Some text}{image}
\keyfig{lw=1,f,r=90,c={Second subfigure},
    l=fig:secondsubfig,
    t=Lots of lots of lots of lots of text.}
    {image2}
\begin{keyfloats}{1}
\keyfig{lw=1,f,c={Third subfigure},l=fig:thirdsubfig}{image}
\keytab{c={Fourth subfigure},l=fig:fourthsubfig}{\testtable}
\keyfig{lw=.5,f,c={Fifth subfigure},l=fig:fifthsubfig}{image}
\end{keyfloats}
\end{keysubfigs}
```

Result:

Figure 15

Figures 15a to 15e are in the fig. 15 `keysubfigs` environment. The `\keysubtabs` environment is similar. Mixed types have the type of their container, as shown with fig. 15d.

Subfloats are associated floats (a, b, . . .) collected together into one common float (the enclosing `\keysubfigs` or `\keysubtabs` environment). The enclosing float can have its own caption (call "Sub-Figures" in the example), which appears in the LOF/LOT, and also a label. Each subfloat can have its own caption and label as well, but the subcaption does not appear in the LOF/LOT.

⚠ mixed subfloats

All subfloats are forced to have the same type as its containing float. A table inside a figure will be labeled as a figure, for example. This avoids miss-labeling as each subfloat must clearly be identified as a child of its containing float.

⚠ nested subfloats

`\keysubfigs` and `\keysubtabs` may not be used inside the `keyfloats` environment, and cannot be nested inside each other. (No subfloat `12aa`, `12ab`, `12ba`, etc.)

nested keyfloats

The `keyfloats` environment may be used inside `\keysubfigs` or `\keysubtabs` to gather subfloats together, such as the three right-most figures in fig. 15.

Subfloats may be located `H`:

`\begin{keysubfigs}[H]{3}{`key/vals . . . `}`

Subfloats may be starred to span both columns in a two-column format:

`\begin{keysubfigs}*{2}{`key/vals . . . `}`

Example 17: Subtables [H] — `keysubtabs` environment

Code:

```
\begin{keysubtabs}[H]{2}{c=Subtables [H],l=tab:subtabs}
\keytab{c={First subtable},l=fig:firstsubtab}{\testtable}
\keytab{c={Second subtable},l=fig:secondsubtab}{\testwidetable}
\end{keysubtabs}
```

Result:

Table 11

Table 11: Subtables [H]

a: First subtable

A	B
C	D

b: Second subtable

A	B	C
D	E	F

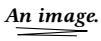


Figure 16: Figure to be continued

Figure 16: . . . continued

2.5.4 Continued Floats

The `cont` key may be used to generate a "continued" float. The continued float receives the same number as the previous float, and it is assumed that they are the same float, except that they are separated for some reason such as size on the page.

The label may be placed in a continued float, and will still receive the same float number as the prior non-continued float.

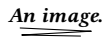
Example 18: Continued figure

Code:

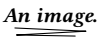
```
\begin{keyfloats}{2}
\keyfig{,c=Figure to be continued}{image}
\keyfig{c={\dots continued},cont,l=fig:firstcontinued}{image2}
\end{keyfloats}
```

Result:

Figure 16



a: First of a set



b: Second of a set

Figure 17: A set of figures

c: Third of a set

d: Fourth of a set

Figure 17: . . . continued

2.5.5 Continued Subfloats

The `keysubfigs` and `keysubtabs` environments may also be given the `cont` key. The containing environment's float receives the same number as the previous float (presumably another subfloat container).

Example 19: Continued subfloats

Code:

```
\begin{keysubfigs}{2}{c={A set of figures},l=fig:continuedfigures}
\keyfig{c={First of a set},l=fig:contfirst}{image}
\keyfig{c={Second of a set},l=fig:contsecond}{image}
\end{keysubfigs}
\begin{keysubfigs}{2}{c={\dots continued},cont}
\keyfig{c={Third of a set},l=fig:contthird}{image2}
\keyfig{c={Fourth of a set},l=fig:contfourth}{image2}
\end{keysubfigs}
```

Result:

Figure 17

2.5.6 Margin Floats

When a keyfloat is located [M], it will be placed in the margin.

Cls `tufte-book`

When the `tufte-book` class is used, its `marginfigure` or `margintable` environments will be used, otherwise keyfloat provides environments of the same name and uses those instead.

Example 20: The `marginfigure` environment

Code:

```
\begin{marginfigure}
\centering
\includegraphics[width=.75\linewidth]{image}

Some text added by hand.
\caption{A \env{marginfigure}}
\label{fig:marginfigure}
\end{marginfigure}
```

Result:

Figure 18

Some text added by hand.

Figure 18: A `marginfigure`

Example 21: The `margintable` environment

Code:

```
\begin{margintable}
\centering
\testwidetable
\caption{A \env{margintable}}
\label{fig:margintable}
\end{margintable}
```

Result:

Table 12

A	B	C
D	E	F

Table 12: A `margintable`

Example 22: Using `\keyfig[M]`

Code:

```
\keyfig[M]{c={A \cs{keyfig}\texttt{[M]}},l=fig:keyfigm,ft,
    t=Additional text.
    Text text text text text text.

    More paragraphs.
}{image2}
```

Result:

Figure 19

Additional text. Text text text text text.

More paragraphs.

Figure 19: A `\keyfig[M]`

Example 23: Using `keytable[M]` and an offset

Code:

```
\begin{keytable}[M]{c={A \env{keytable}\texttt{[M]}},
    l=tab:keytablem,mo=-.9in}
\testwidetable
\end{keytable}
```

Result:

Table 13

Table 13: A `keytable[M]`

A	B	C
D	E	F

margin float offset

A negative offset was used to shift the table upwards to the top of the example.

distance between floats

To set the minimum-allowed distance between `\marginpar`s and margin floats:

```
\setlength{\marginparpush}{3ex}
```

2.5.7 Wrapped Floats

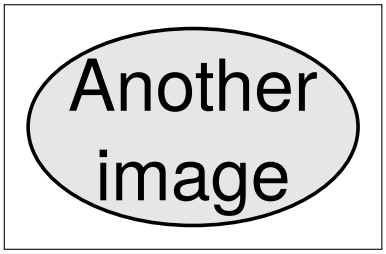
Example 24: Using `\keyfig[W]` and `\keytab[W]`

Code:

```
\keyfig[W]{c={A \cs{keyfig}\texttt{[W]}},
    l=fig:keyfigw,ft,lw=.4,wp=I,
    t={.4\cs{linewidth} wide, placed \texttt{I}.}
}{image2}
\lipsum[1]
\keytab[W]{c={A \cs{keytab}\texttt{[W]}},l=tab:keytabw,w=.75in,
}{\testtable}
\lipsum[2]
```

Result:

Figure 20 and table 14



.4`\linewidth` wide, placed `I`.

Figure 20: A `\keyfig[W]`

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Table 14: A `\keytab[W]`

A	B
C	D

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Example 25: Using `\keyfigbox[W]` and `\keyparbox[W]`

Code:

```
\keyfigbox[W]{c={A \cs{keyfigbox}\texttt{[W]}},
    l=fig:keyfigboxw,f,lw=.25,wp=I,
    t=Text text text text text text text text text
}{The contents.}
\lipsum[1]
\keyparbox[W]{w=1in}{A \cs{keyparbox}[W] and some more text.}
\lipsum[2]
```

Result:

Figure 21 and the `\keyparbox`.

Text text text text text text text text text text

Figure 21: A `\keyfigbox[W]`

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

A `\keyparbox`[W] and some more text.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Example 26: Using `\keyfigure[W]` and `\keytable`[W]

Code:

```
\begin{keyfigure}[W]{c={A \cs{keyfigure}\texttt{[W]}},
    l=fig:keyfigurew,f,w=1.5in}
This is a keyfigure.
\end{keyfigure}
\lipsum[1]

\begin{keytable}[W]{c={A \env{keytable}\texttt{[W]}},
    l=tab:keytablew,w=2in,wp=L,tc=Placed \texttt{L} and 2in wide.}
\testwidetable
\end{keytable}
\lipsum[2]
```

Result:

Figure 22 and table 15

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

This is a keyfigure.

Figure 22: A `\keyfigure[W]`

Table 15: A `keytable[W]`

A	B	C
D	E	F

Placed `L` and 2in wide.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Example 27: Using \keywrap with a \keyfig

Code:

```
\begin{itemize}
\item First item.
    Several lines of text text text text text
    text text text text text text text text.
\item \begin{keywrap}{.3\linewidth}{\keyfig{%
      lw=1,c={Keywrap with \cs{keyfig}},l=fig:keywrapfig%
    }{image}}
        Second item.
        Several lines of text text text text text
        text text text text text text text text text
        text text text text text text text.

        These paragraphs are inside the \texttt{keywrap}.
        A vertical gap appears below if the text is not enough to
        fill the space next to the \cs{keyfig}.
    \end{keywrap}
    Outside the \env{wrapfig},\margintag{notes}\
    but still in the second item.
    There is no elegant way to place only part of a paragraph
    inside a \env{keywrap}, and attempting to do so requires
    manually removing the vertical paragraph skip.
\item Third item.
\end{itemize}
```

Result:

Figure 23

- First item. Several lines of text text text text text text text text text text text text text.
- Second item. Several lines of text text text text text text text text text text text text text text text text text text text text text text text text text.

  These paragraphs are inside the `keywrap`. A vertical gap appears below if the text is not enough to fill the space next to the `\keyfig`.

Figure 23: Keywrap with \keyfig

notes

  Outside the `wrapfig`, but still in the second item. There is no elegant way to place only part of a paragraph inside a `keywrap`, and attempting to do so requires manually removing the vertical paragraph skip.

- Third item.

Figure 24: Custom-framed image

A loosely-framed box.

Figure 25: Custom loosely-framed box

2.5.8 Custom Frames

Example 28: Custom frames with mdframed

Code:

```
\renewcommand{\KFLTtightframe}[1]{%
\begin{minipage}{\KFLTimageboxwidth}
\begin{mdtightframe}%
#1
\end{mdtightframe}%
\end{minipage}
}
\setlength{\KFLTtightframewidth}{1pt}

\renewcommand{\KFLTlooseframe}[1]{%
\begin{mdlooseframe}[leftmargin=1.5in,rightmargin=1.5in]%
#1
\end{mdlooseframe}%
}
\setlength{\KFLTlooseframewidth}{4pt}

\keyfig{ft,c=Custom-framed image,l=fig:customframe,r=90}{image}
\keyfigbox{f,c=Custom loosely-framed box,
    l=fig:customlooseframe}{A loosely-framed box.}
```

Result:

Figures 24 and 25

Pkg `mdframed`

⚠ mdframed width

Example 28 shows custom frames created with the mdframed package along with tikz. Note that mdframed uses the full `\linewidth` even if the left/right margins are explicitly set, which causes extra vertical space when rotated. Because of this, the framed object is enclosed inside a minipage whose width is precomputed based on the object itself, then set in `\KFLTimageboxwidth`. Any shadow may fall

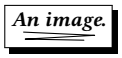


Figure 26: Custom shadow

A loosely-framed shadow box.

Figure 27: Custom loosely-framed shadow

outside this box.

See section 2.6.1 for more details.

Example 29: Custom shadows with fancybox

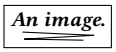
Code:

```
\renewcommand{\KFLTtightframe}[1]{%
\setlength{\fboxrule}{.4pt}
\setlength{\fboxsep}{0pt}
\setlength{\shadowsize}{2pt}
\shadowbox{#1}%
}
\setlength{\KFLTtightframewidth}{0.4pt}

\renewcommand{\KFLTlooseframe}[1]{%
\setlength{\fboxrule}{.4pt}
\setlength{\fboxsep}{3pt}
\setlength{\shadowsize}{2pt}
\shadowbox{#1}%
}
\setlength{\KFLTlooseframewidth}{3.4pt}

\keyfig{ft,c=Custom shadow,l=fig:customshadow}{image}
\keyfigbox{f,c=Custom loosely-framed shadow,lw=.5,
    l=fig:customlooseshadow}{A loosely-framed shadow box.}
```

Result:
Figures 26 and 27

Pkg `fancybox` Example 29 shows custom shadow frames created with the fancybox package. This combination respects `lw` and `w`.

See section 2.6.1 for more details.



Mr. First Last III

About the illustration.

Figure 28: Artist's name — image

Some text, a quotation, a TikZ diagram — anything not an image file.

Mr. Last

Figure 29: Artist's name — arbitrary contents

2.5.9 Artist's Name

Example 30: Artist's name — image

Code:

```
\keyfig{ft,ap=Mr.,af=First,al=Last,as={~III},
tc={\textit{About the illustration.}},
c=Artist's name --- image,l=fig:artist}{image}
```

Result:
Figure 28

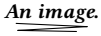
Example 31: Artist's name — arbitrary contents

Code:

```
\tdnameright
\begin{keyfigure}{f,ap=Mr.,al=Last,
c=Artist's name --- arbitrary contents,l=fig:artistpar}
\centering Some text, a quotation, a TikZ\ diagram ---
anything not an image file.
\end{keyfigure}
\tdnamecenter
```

Result:
Figure 29

The artist's name and optional prefix/suffix are printed below the figure, and an index entry is made for the name in (Last, First) format, or (Last) if there is no first name. If the tocdata package is loaded, the artist's name is also added to



a: Artist's First Work

Commentary about the work.

b: Artist's Second Work

Prefix First Last, Suffix

Some fully-justified text just for illustrative purposes, in case you have use for long explanations. This text may be the full `\linewidth` in size.

Multiple paragraphs of text are allowed.

Figure 30: Artist's collection

the List of Figures, and the tocdata `\tdname`... macros may be used to align the name.

Example 32: Subfloats with an artist

Code:

```
\begin{keysubfigs}{2}{
    c=Artist's collection, l=fig:artistcollection,
    t={Some fully-justified text just for illustrative purposes,
    in case you have use for long explanations.
    This text may be the full \cs{linewidth} in size. \par
    Multiple paragraphs of text are allowed.},
    ap=Prefix,af=First,al=Last,as={, Suffix}
}
    \keyfig{c=Artist's First Work}{image}
    \keyfig{c=Artist's Second Work,
        tc={Commentary about the work.}}{image2}
\end{keysubfigs}
```

Result:

Figure 30

A group of figures may be placed into a subfloat container, which may have its own artist keys and additional text. Furthermore, each subfloat inside the collection may also have its own artist tags and additional text.

2.6 Customization

2.6.1 Custom Frames

There are two user-redefinable framing macros:
`\KFLTtightframe` and `\KFLTlooseframe`

A float's contents are placed into a box, which is passed to either of these two macros depending on the key `f` or `tf`.

Each macro takes one argument and frames it.

Each macro has an associated LaTeX length:
`\KFLTtightframewidth` and `\KFLTlooseframewidth`

These lengths must be redefined to the expected total frame width, equal to the frame thickness plus separation.

The defaults definitions are:

```
\newcommand{\KFLTtightframe}[1]{%
\setlength{\fboxsep}{0pt}%
\setlength{\fboxrule}{.4pt}%
\fbox{#1}%
}
\setlength{\KFLTtightframewidth}{.4pt}

\newcommand{\KFLTlooseframe}[1]{%
\setlength{\fboxsep}{3pt}%
\setlength{\fboxrule}{.4pt}%
\fbox{#1}%
}
\setlength{\KFLTlooseframewidth}{3.4pt}
```

See example 28 for an example created with the mdframed package, and example 29 for an example created with the fancybox package.

2.6.2 Distance between Floats and Rows

rows too close/far

To spread out the distance between floats and/or rows of floats on a busy page, the following adjustments may be made. The values used in this documentation are:

```
\setlength{\floatsep}{5ex plus 1ex minus 1ex}
\setlength{\dblfloatsep}{5ex plus 1ex minus 1ex}
```

2.6.3 Formatting the Captions

To modify the typesetting of the captions, see the caption package. The settings used in this documentation are:

```
% default applied to margin floats:
\captionsetup{labelfont={small,bf},textfont={small,bf}}

\captionsetup[figure]{
 style=default, justification=centering,
 margin=0pt, parskip=0pt, skip=2ex,
 labelfont={small,bf},textfont={small,bf}
}

\captionsetup[table]{
 style=default, justification=centering,
 margin=0pt, parskip=0pt, skip=1ex,
 labelfont={small,bf},textfont={small,bf}
}

\captionsetup[subfigure]{
 style=default, justification=centering,
 margin=0pt, parskip=0pt, skip=2ex,
 labelfont={small},textfont={small}
}

\captionsetup[subtable]{
 style=default, justification=centering,
 margin=0pt, parskip=0pt, skip=1ex,
 labelfont={small},textfont={small}
}
```

3 Code

3.1 Required Packages

Pkg **etoolbox** v2.6 or later for `\BeforeBeginEnvironment`, `\AfterEndEnvironment`

```
\RequirePackage{etoolbox}[2011/01/03]%

\RequirePackage{xparse}

\RequirePackage{xifthen}
```

Pkg **keyval** Key processing:

```
\RequirePackage{xkeyval}
```

Pkg **graphicx** For `\includegraphics` and rotating:

```
\RequirePackage{graphicx}
```

Pkg **caption** Handles all caption-related functions:

```
\RequirePackage{caption}[2010/10/31]% v3.2 to support \phantomcaption
```

Pkg **subcaption** Derived from caption, used to handle subfloats:

```
\RequirePackage{subcaption}
```

Pkg **calc** Used to compute box width minus frame sep and width.

```
\RequirePackage{calc}
```

Pkg **rotating** Provides rotation via the `turn` environment:

```
\RequirePackage{rotating}
```

Pkg **placeins** Provides

to process existing floats before adding new ones.

```
\RequirePackage{placeins}
```

Pkg **wrapfig** Provides figure wrapping code.

```
\RequirePackage{wrapfig}
```

Package error if floatrow was loaded:

```
\@ifpackageloaded{floatrow}
{
\PackageError{keyfloat}
{The keyfloat conflicts with the floatrow package.
Remove floatrow to use keyfloat.}
{Keyfloat uses the caption and subcaption packages to
provide similar functionality to floatrow.}
}
{}
```

Pkg `gettitlestring` Used by hyperref and nameref.

Expand names used in titles:

```
\PassOptionsToPackage{expand}{gettitlestring}
```

Rows of floats are created by a simple `minipage` environment, instead of relying on a preexisting package. This proved to be advantageous when support was added for multiple rows in one environment.

3.2 In-line Figures and Tables

These macros are commonly used by others.

Env `tablehere` Place a table exactly [H].

```
\ProvideDocumentEnvironment{tablehere}{}
  {\bigbreak\noindent\minipage{\linewidth}\def\@captype{table}}
  {\endminipage\bigbreak}
```

Env `figurehere` Place a figure exactly [H].

```
\ProvideDocumentEnvironment{figurehere}{}
  {\bigbreak\noindent\minipage{\linewidth}\def\@captype{figure}}
  {\endminipage\bigbreak}
```

3.3 Row Counting and Control

Used to count position and wrap at end of each row.

Ctr `KFLT@numcols` Columns per row.

```
\newcounter{KFLT@numcols}
```

Ctr `KFLT@thiscol` Column currently processing. `0` if not yet in a keyfloats or subfloat.

```
\newcounter{KFLT@thiscol}
```

Len `\KFLT@rowboxwidth` How wide is each box in the row.

```
\newlength{\KFLT@rowboxwidth}
```

3.4 Float Key Handling

Bool `KFLT@cont` Continued float?

```
\newboolean{KFLT@cont}
```

Key `main` `cont` Continued float?

```
\define@key{KFLT@keys}{cont}[true]{\setboolean{KFLT@cont}{#1}}
```

`\KFLT@c` Caption storage

```
\newcommand{\KFLT@c}{}
```

Bool `KFLT@cstar` Starred caption?

```
\newboolean{KFLT@cstar}
```

Key `main` `c` Caption

```
\define@key{KFLT@keys}{c}%
{\renewcommand{\KFLT@c}{#1}\setboolean{KFLT@cstar}{false}}
```

Key `main` `cstar` Caption starred?

```
\define@key{KFLT@keys}{cstar}%
{\renewcommand{\KFLT@c}{#1}\setboolean{KFLT@cstar}{true}}
```

Key `main` `sc` Short caption

```
\define@key{KFLT@keys}{sc}{%
\renewcommand{\KFLT@sc}{#1}%
\setboolean{KFLT@scgiven}{true}%
}
```

`\KFLT@sc` Short caption storage

```
\newcommand{\KFLT@sc}{}
```

Bool `KFLT@scgiven` Was a short caption given?

```
\newboolean{KFLT@scgiven}
```

`\KFLT@type` Float type: "`figure`", "`table`"

```
\newcommand*{\KFLT@type}{}
```

`\KFLT@listtype` List type: "`lof`", "`lot`"

```
\newcommand*{\KFLT@listtype}{}
```

Key `main` `l` Label

```
\define@key{KFLT@keys}{l}{\renewcommand{\KFLT@l}{#1}}
```

`\KFLT@l` Label storage

```
\newcommand*{\KFLT@l}{}
```

For the artist/author keys:

Key `main` `ap` Artist prefix

```
\define@key{KFLT@keys}{ap}{\renewcommand{\KFLT@ap}{#1}}
```

`\KFLT@ap` Storage for artist prefix

```
\newcommand*{\KFLT@ap}{}
```

Key `main` `af` Artist first name

```
\define@key{KFLT@keys}{af}{\renewcommand{\KFLT@af}{#1}}
```

`\KFLT@af` Storage for artist first name

```
\newcommand*{\KFLT@af}{}
```

Key `main` `al` Artist last name

```
\define@key{KFLT@keys}{al}{\renewcommand{\KFLT@al}{#1}}
```

`\KFLT@al` Storage for artist last name

```
\newcommand*{\KFLT@al}{}
```

Key `main` `as` Artist suffix

```
\define@key{KFLT@keys}{as}{\renewcommand{\KFLT@as}{#1}}
```

`\KFLT@as` Storage for artist suffix

```
\newcommand*{\KFLT@as}{}
```

`\KFLT@textalign` Storage for text alignment.

Used for the additional text in the float.

```
\newcommand*{\KFLT@textalign}{}
```

`\KFLT@t` Additional text storage

Used for the additional text in the float.

```
\newcommand{\KFLT@t}{}
```

Create replacement macros in case tocdata is not loaded:

```
\providecommand{\tdtextjustify}{}
\providecommand{\tdtextcenter}{}
\providecommand{\tdtextleft}{}
\providecommand{\tdtextright}{}
\providecommand{\tdnamejustify}{}
\providecommand{\tdnamecenter}{}
\providecommand{\tdnameleft}{}
\providecommand{\tdnameright}{}
```

Key `main` `t` Additional text, justified alignment.

```
\define@key{KFLT@keys}{t}{
\renewcommand{\KFLT@t}{#1}
\renewcommand{\KFLT@textalign}{}
\tdtextjustify
}
```

Key `main` `tc` Additional text, centered alignment.

```
\define@key{KFLT@keys}{tc}{
```

```
\renewcommand{\KFLT@t}{#1}
\renewcommand{\KFLT@textalign}{\centering}
\tdtextcenter
}
```

Key main **tr** Additional text, aligned to the right.

```
\define@key{KFLT@keys}{tr}{
\renewcommand{\KFLT@t}{#1}
\renewcommand{\KFLT@textalign}{\raggedleft}
\tdtextright
}
```

Key main **tl** Additional text, aligned to the left.

```
\define@key{KFLT@keys}{tl}{
\renewcommand{\KFLT@t}{#1}
\renewcommand{\KFLT@textalign}{\raggedright}
\tdtextleft
}
```

\KFLT@i Image filename storage

```
\newcommand*{\KFLT@i}{}
```

Key main **lw** Fraction of `\linewidth`

```
\define@key{KFLT@keys}{lw}{%
\renewcommand{\KFLT@lw}{#1}%
\setlength{\KFLT@w}{0pt}%
}
```

\KFLT@lw Fraction of linewidth storage: "`.5`"

```
\newcommand*{\KFLT@lw}{}
```

Key main **w** Fixed width

```
\define@key{KFLT@keys}{w}{%
\setlength{\KFLT@w}{#1}%
\renewcommand{\KFLT@lw}{}%
}
```

\KFLT@w Width storage: "3cm"

```
\newlength{\KFLT@w}
```

Key main `h` Fixed height

```
\define@key{KFLT@keys}{h}{\setlength{\KFLT@h}{#1}}
```

`\KFLT@h` Height storage: “2in”

```
\newlength{\KFLT@h}
```

Key main `s` Scale

```
\define@key{KFLT@keys}{s}{\renewcommand{\KFLT@s}{#1}}
```

`\KFLT@s` Scale storage: “3”

```
\newcommand*{\KFLT@s}{1}
```

Key main `r` Angle. 90 is counter-clockwise 90 degrees.

```
\define@key{KFLT@keys}{r}{\renewcommand{\KFLT@r}{#1}}
```

`\KFLT@r` Angle storage: “90”

```
\newcommand*{\KFLT@r}{0}
```

Key main `f` Frame the image with `\KFLTlooseframe`.

```
\define@key{KFLT@keys}{f}[true]{\setboolean{KFLT@f}{#1}}
```

Bool `KFLT@f` Frame the image?

```
\newboolean{KFLT@f}
```

Key main `ft` Tightly frame the image using `\KFLTtightframe`. This is useful for photographs, or diagrams which already have built-in margins.

```
\define@key{KFLT@keys}{ft}[true]{\setboolean{KFLT@ft}{#1}}
```

Bool `KFLT@ft` Tightly frame the image?

```
\newboolean{KFLT@ft}
```

Key main `stretch` Set `\arraystretch` inside the table environment.

```
\define@key{KFLT@keys}{stretch}{\renewcommand{\KFLT@stretch}{#1}}
```

`\KFLT@stretch` Storage for `\arraystretch`.

```
111 \newcommand*{\KFLT@stretch}{1}
```

Key `main` `mo` Set vertical offset for a margin float.

```
112 \define@key{KFLT@keys}{mo}{\setlength{\KFLT@mo}{#1}}
```

`\KFLT@mo` Storage for the vertical margin offset.

```
113 \newlength{\KFLT@mo}
```

Key `main` `wp` Set wrap placement for a wrapped float.

See table 3 on page 14.

```
114 \define@key{KFLT@keys}{wp}{\renewcommand{\KFLT@wp}{#1}}
```

`\KFLT@wp` Storage for the wrap placement.

```
115 \newcommand{\KFLT@wp}{O}
```

Key `main` `va` Set vertical alignment of the outermost minipage container.

```
116 \define@key{KFLT@keys}{va}{\renewcommand{\KFLT@va}{#1}}
```

`\KFLT@va` Storage for the vertical alignment.

```
117 \newcommand{\KFLT@va}{c}
```

3.5 Nesting Control

Ctr `KFLT@keyfloatdepth` Depth inside a keyfigs environment

```
118 \newcounter{KFLT@keyfloatdepth}
119 \setcounter{KFLT@keyfloatdepth}{0}
```

Bool `KFLT@inkeysubfloats` Inside a keysubfigs environment?

```
120 \newboolean{KFLT@inkeysubfloats}
121 \setboolean{KFLT@inkeysubfloats}{false}
```

3.6 Subfloat Key Handling

These keys are for the container holding a collection of subfigures.

Bool `KFLT@subgrpcont` Continued float?

```
\newboolean{KFLT@subgrpcont}{}
```

Key subfloat container `cont` Continued float

```
\define@key{KFLT@subgrpkeys}{cont}[true]{%
\setboolean{KFLT@subgrpcont}{#1}%
}
```

`\KFLT@subgrpc` Sub-caption storage

```
\newcommand{\KFLT@subgrpc}{}
```

Bool `KFLT@subgrpcstart` Sub-caption starred?

```
\newboolean{KFLT@subgrpcstar}
```

Key subfloat container `c` Caption

```
\define@key{KFLT@subgrpkeys}{c}
{\renewcommand{\KFLT@subgrpc}{#1}\setboolean{KFLT@subgrpcstar}{false}}
```

Key subfloat container `cstar` Starred caption?

```
\define@key{KFLT@subgrpkeys}{cstar}
{\renewcommand{\KFLT@subgrpc}{#1}\setboolean{KFLT@subgrpcstar}{true}}
```

Key subfloat container `sc` Short caption

```
\define@key{KFLT@subgrpkeys}{sc}{%
\renewcommand{\KFLT@subgrpsc}{#1}%
\setboolean{KFLT@subgrpscgiven}{true}%
}
```

`\KFLT@subgrpsc` Sub-shortcaption storage

```
\newcommand{\KFLT@subgrpsc}{}
```

Bool `KFLT@subgrpscgiven` Sub-shortcaption was given?

```
\newboolean{KFLT@subgrpscgiven}
```

`\KFLT@subgrptype` Subfloats collection type storage: "`figure`", "`table`"

```
\newcommand*{\KFLT@subgrptype}{}
```

`\KFLT@subgrptype` Subfloats collection list type storage: "`lof`", "`lot`"

```
\newcommand*{\KFLT@subgrplisttype}{}
```

`\KFLT@setsubgrpfigure` Set to figure type

```
\newcommand*{\KFLT@setsubgrpfigure}{%
\renewcommand{\KFLT@subgrptype}{figure}%
\renewcommand{\KFLT@subgrplisttype}{lof}%
}
```

`\KFLT@setsubgrptable` Set to table type

```
\newcommand*{\KFLT@setsubgrptable}{%
\renewcommand{\KFLT@subgrptype}{table}%
\renewcommand{\KFLT@subgrplisttype}{lot}%
}
```

Key `subfloat container` `l` Label

```
\define@key{KFLT@subgrpkeys}{l}{\renewcommand{\KFLT@subgrpl}{#1}}
\newcommand*{\KFLT@subgrpl}{}
```

`\KFLT@subgrptextalign` Storage for text alignment.

Used for the additional text in the float.

```
\newcommand*{\KFLT@subgrptextalign}{}
```

`\KFLT@subgrpt` Additional text storage

Used for the additional text in the float.

```
\newcommand{\KFLT@subgrpt}{}
```

Key `subfloat container` `t` Additional text — full justification

```
\define@key{KFLT@subgrpkeys}{t}{
\renewcommand{\KFLT@subgrpt}{#1}
\renewcommand{\KFLT@subgrptextalign}{}
\tdtextjustify
}
```

Key subfloat container t Additional text — center justification

```
\define@key{KFLT@subgrpkeys}{tc}{
\renewcommand{\KFLT@subgrpt}{#1}
\renewcommand{\KFLT@subgrptextalign}{\centering}
\tdtextcenter
}
```

Key subfloat container t Additional text — aligned left

```
\define@key{KFLT@subgrpkeys}{tl}{
\renewcommand{\KFLT@subgrpt}{#1}
\renewcommand{\KFLT@subgrptextalign}{\raggedright}
\tdtextleft
}
```

Key subfloat container t Additional text — aligned right

```
\define@key{KFLT@subgrpkeys}{tr}{
\renewcommand{\KFLT@subgrpt}{#1}
\renewcommand{\KFLT@subgrptextalign}{\raggedleft}
\tdtextright
}
```

For the tocdata package:

Key subfloat container ap Artist prefix

```
\define@key{KFLT@subgrpkeys}{ap}{\renewcommand{\KFLT@subgrpap}{#1}}
```

\KFLT@subgrpap Storage for artist prefix

```
\newcommand*{\KFLT@subgrpap}{}
```

Key subfloat container af Artist first name

```
\define@key{KFLT@subgrpkeys}{af}{\renewcommand{\KFLT@subgrpaf}{#1}}
```

\KFLT@subgrpaf Storage for artist first name

```
\newcommand*{\KFLT@subgrpaf}{}
```

Key subfloat container al Artist last name

```
\define@key{KFLT@subgrpkeys}{al}{\renewcommand{\KFLT@subgrpal}{#1}}
```

\KFLT@subgrpal Storage for artist last name

```
\newcommand*{\KFLT@subgrpal}{}
```

Key subfloat container as Artist suffix

```
\define@key{KFLT@subgrpkeys}{as}{\renewcommand{\KFLT@subgrpas}{#1}}
```

\KFLT@subgrpas Storage for artist suffix

```
\newcommand*{\KFLT@subgrpas}{}
```

3.7 Computing Image Width

Len \KFLT@imagewidth Computed width of the image

```
\newlength{\KFLT@imagewidth}
```

Len \KFLT@boxwidth Computed width of the container box

```
\newlength{\KFLT@boxwidth}
```

\KFLT@findwidths Figure out how wide to make an image and its container

```
\newcommand*{\KFLT@findwidths}{%
```

Default to a box of full \linewidth minus the potential frame:

```
\ifthenelse{\boolean{KFLT@ft}}% tight frame?
{\setlength{\KFLT@boxwidth}{\linewidth - 2\KFLTtightframewidth}}%
{% not tight frame
\ifthenelse{\boolean{KFLT@f}}% loose frame?
{\setlength{\KFLT@boxwidth}{\linewidth - 2\KFLTlooseframewidth}}%
{\setlength{\KFLT@boxwidth}{\linewidth}}% no frame
}% not tight frame
```

Several width options exist. First see if width was given:

```
\ifthenelse{\dimtest{\KFLT@w}{>}{0pt}}%
```

Width was given:

```
{\setlength{\KFLT@imagewidth}{\KFLT@w}}%
{% width not given
```

Use full \linewidth or only a fraction:

```
\ifcsempty{\KFLT@lw}%
{\setlength{\KFLT@imagewidth}{\KFLT@boxwidth}}%
{\setlength{\KFLT@imagewidth}{\KFLT@lw\KFLT@boxwidth}}%
}% width not given
}
```

3.8 Framing and Rotation

A user-redefinable macro and length to tightly frame the contents.

\KFLTtightframe may be redefined to a macro which frames its contents. \KFLTtightframewidth should be redefine to the total width of the new frame and its separation.

\KFLT@tightframe {⟨*contents*⟩}

```
\newcommand{\KFLTtightframe}[1]{%
\setlength{\fboxsep}{0pt}%
\setlength{\fboxrule}{.4pt}%
\fbox{#1}%
}

```

Len \KFLTtightframewidth Combined width of the frame and separation.

```
\newlength{\KFLTtightframewidth}
\setlength{\KFLTtightframewidth}{.4pt}
```

\KFLTlooseframe {⟨*contents*⟩}

A user-redefinable macro and length to loosely frame the contents.

\KFLTlooseframe may be redefined to a macro which frames its contents. \KFLTlooseframewidth should be redefine to the total width of the new frame and its separation.

```
\newcommand{\KFLTlooseframe}[1]{%
\setlength{\fboxsep}{3pt}%
\setlength{\fboxrule}{.4pt}%
\fbox{#1}%
}
```

Len \KFLTlooseframewidth Combined width of the frame and separation.

```
\newlength{\KFLTlooseframewidth}
\setlength{\KFLTlooseframewidth}{3.4pt}
```

`\KFLT@frame` `{`⟨*contents*⟩`}`

Frames the contents according to the `f` key. To be nested for further processing.

```
\newcommand{\KFLT@frame}[1]
{%
\ifthenelse{\boolean{KFLT@ft}}%
{\KFLTtightframe{#1}}%
{% not tightframe
\ifthenelse{\boolean{KFLT@f}}%
{\KFLTlooseframe{#1}}%
{#1}% no frame
}% not looseframe
}
```

`KFLT@findenvboxwidth` Figures the width of the contents of `\KFLT@envbox` plus the frame:

```
\newcommand{\KFLT@findenvboxwidth}{%
\settowidth{\KFLTimageboxwidth}{\usebox{\KFLT@envbox}}%
\ifthenelse{\boolean{KFLT@ft}}%
{\addtolength{\KFLTimageboxwidth}{2\KFLTtightframewidth}}%
{% not tightframe
\ifthenelse{\boolean{KFLT@f}}%
{\addtolength{\KFLTimageboxwidth}{2\KFLTlooseframewidth}}%
{}% no frame
}% not looseframe
}
```

3.9 A Graphics Image from a File

`\KFLT@onefigureimage` Create a stand-alone figure with an image.

```
\NewDocumentCommand{\KFLT@onefigureimage}{}
{%
```

Several possible combinations of linewidth, width, and height are available, and each is treated separately. Scaling and width/height are done first, then framing, then rotation.

```
\begin{lrbox}{\KFLT@envbox}%
```

Handle the `lw` key. If `lw` is used, width and height are ignored.

```
\ifthenelse{\NOT\equal{\KFLT@lw}{}}%
{\includegraphics%
[scale=\KFLT@s,width=\KFLT@imagewidth]{\KFLT@i}}%
{% not linewidth
```

Handle the `w` key, which may be used along with the `h` key:

```
\ifthenelse{\dimtest{\KFLT@w}{>}{0pt}}%
{% width is given
\ifthenelse{\dimtest{\KFLT@h}{>}{0pt}}%
```

Width and height are both given:

```
{% w and h
\includegraphics%
[scale=\KFLT@s,%
width=\KFLT@imagewidth,height=\KFLT@h]{\KFLT@i}%
}% w and h
```

Only width:

```
{% only w
\includegraphics%
[scale=\KFLT@s,width=\KFLT@imagewidth]{\KFLT@i}%
}% only w
}% width is given
```

Width was not given, so maybe handle `h` alone:

```
{% width is not given
\ifthenelse{\dimtest{\KFLT@h}{>}{0pt}}%
```

`h` was given:

```
{\includegraphics%
[scale=\KFLT@s,height=\KFLT@h]{\KFLT@i}}%
```

If none were given, use the image's natural size:

```
{\includegraphics%
[scale=\KFLT@s]{\KFLT@i}}%
}% width is not given
}% not linewidth
\end{lrbox}%
\unskip%
\KFLT@findenvboxwidth%
\begin{turn}{\KFLT@r}%
```

Encapsulate the frame in case the custom frame commands used pars:

```
% \begin{minipage}{\KFLTimageboxwidth}%
\KFLT@frame{\usebox{\KFLT@envbox}}%
% \end{minipage}%
\unskip%
\end{turn}%
}
```

3.10 Printing the Caption

`\KFLT@captioniftype` {⟨*figure or table*⟩} {⟨*{} or subgrp*⟩}

Create a caption only if is of this float type.

The second argument is `{}` if a regular float, or `subgrp` if `\keysubfigs` or `\keysubtabs`.

```
\newcommand*{\KFLT@captioniftype}[2]{%
\ifthenelse{\equal{\csname KFLT@#2type\endcsname}{#1}}%
{\KFLT@caption{#2}}%
{}%
}
```

`\KFLT@dosimplecaption` {⟨*star?*⟩} {⟨*short cap or -NO VALUE-*⟩} {⟨*caption*⟩}

Calls `\caption` depending on several combinations of star and short captions being given.

```
\NewDocumentCommand{\KFLT@dosimplecaption}{m m m}
{%
\unskip%
\IfBooleanTF{#1}% star?
{% star
\IfValueTF{#2}{\caption*[#2]{#3}}{\caption*{#3}}%
}% star
{% no star
\IfValueTF{#2}{\caption[#2]{#3}}{\caption{#3}}%
}% no star
}
```

`\KFLT@docaption` * [⟨*short caption*⟩] {⟨*caption*⟩} {⟨*{} or subgrp*⟩}

Depending on whether the tocdata package is present, and an artist is specified, use either `\caption` or `\captionartist`.

The fourth argument is `{}` if a regular float, or `subgrp` if `\keysubfigs` or `\keysubtabs`.

See Table 2 for the possible combinations of the caption-related keys: `c`, `cstar`, and `sc`.

There are two versions, depending on whether tocdata is loaded.

```
\@ifpackageloaded{tocdata}
{% tocdata loaded
```

tocdata is loaded:

```
\NewDocumentCommand{\KFLT@docaption}{s o m m}
{%
```

Is this a figure?

```
\ifthenelse{\equal{\csname KFLT@#4type\endcsname}{figure}}%
{% figure
```

Is the last name empty? Assume no artist if so.

```
\ifcsempty{KFLT@#4al}%
{% figure w/o artist
```

A figure without an artist uses the simple caption.

```
\KFLT@dosimplecaption{#1}{#2}{#3}%
}% figure w/o artist
{% figure with an artist
```

A figure with an artist uses the tocdata `\captionartist` macro, which also creates an index entry.

```
\IfBooleanTF{#1}{% star
\captionartist*[#2]{#3}%
[\csname KFLT@#4t\endcsname]%
[\csname KFLT@#4ap\endcsname]%
{\csname KFLT@#4af\endcsname}%
{\csname KFLT@#4al\endcsname}%
[\csname KFLT@#4as\endcsname]%
}% star
{% no star
\captionartist[#2]{#3}%
[\csname KFLT@#4t\endcsname]%
[\csname KFLT@#4ap\endcsname]%
{\csname KFLT@#4af\endcsname}%
{\csname KFLT@#4al\endcsname}%
```

```
[\csname KFLT@#4as\endcsname]%
}% no star
}% figure with an artist
}% figure
{% not a figure, ignore artist information:
```

If it isn't a figure, ignore artist information and create a simple caption:

```
\KFLT@dosimplecaption{#1}{#2}{#3}%
}% not a figure
}% KFLT@tocdata
}% tocdata loaded
{% no tocdata
\NewDocumentCommand{\KFLT@docaption}{s o m m}
{%
```

If tocdata is not loaded, use a simple caption.

```
\KFLT@dosimplecaption{#1}{#2}{#3}%
```

Create an index entry depending on whether there is a last, first name:

```
\ifcsempty{KFLT@#4al}%
{}% no artist
{% yes artist
\ifcsempty{KFLT@#4af}%
{\index{\csname KFLT@#4al\endcsname}}%
{\index{\csname KFLT@#4al\endcsname, \csname KFLT@#4af\endcsname}}%
}% yes artist
}% KFLT@docaption
}% no tocdata
```

`\KFLT@caption` {⟨*{}* or *subgrp*⟩}

Caption-creation logic.

The argument is `{}` if a regular float, or `subgrp` if `\keysubfigs` or `\keysubtabs`.

See Table 2 for the possible combinations of the caption-related keys: `c`, `cstar`, and `sc`.

```
\newcommand{\KFLT@caption}[1]{%
```

A starred caption is printed but not numbered.

```
\ifthenelse{\boolean{KFLT@#1cstar}}% starred caption?
```

This is a starred caption:

```
{%starred caption
```

A key given as `cstar={}` yields a float with no caption at all.

```
\ifcsempty{KFLT@#1c}% cstar={}?
{}%
```

Non-empty starred caption might have a LOF entry if it has a short caption `sc` key:

```
{% non-empty starred caption
\ifcsempty{KFLT@#1sc}%
```

No `sc` short caption, but there is a `cstar`, so no LOF entry:

```
{}%
```

Both `cstar` and `sc` were given, so add a LOF entry:

```
{% non-empty cstar and sc:
\addcontentsline{\KFLT@listtype}%
{\csname KFLT@#1type\endcsname}{\KFLT@sc}%
}% non-empty cstar and sc
```

`cstar` was given, so create an unnumbered caption:

```
\KFLT@docaption*{\csname KFLT@#1c\endcsname}{#1}%
}%
}% starred caption
```

Unstarred caption `c` was given, so number this float:

```
{% unstarred caption
\ifcsempty{KFLT@#1sc}%
{% no short cap
\KFLT@docaption{\csname KFLT@#1c\endcsname}{#1}%
}% no short cap
{% short cap
\KFLT@docaption[\csname KFLT@#1sc\endcsname]%
{\csname KFLT@#1c\endcsname}{#1}%
}% short cap
```

Optional label:

```
\ifcsempty{KFLT@#1l}%
{}%
{\label{\csname KFLT@#1l\endcsname}}%
}% unstarred caption
}
```

3.11 Defaults for a New Float

`\KFLT@defaults` Defaults all settings before reading the keys.

```
\newcommand*{\KFLT@defaults}{%
\setboolean{KFLT@cont}{false}%
\renewcommand{\KFLT@c}{}%
\setboolean{KFLT@cstar}{false}%
\renewcommand{\KFLT@sc}{}%
\setboolean{KFLT@scgiven}{false}%
\renewcommand{\KFLT@type}{figure}%
\renewcommand{\KFLT@listtype}{lof}%
\renewcommand{\KFLT@l}{}%
\renewcommand{\KFLT@ap}{}%
\renewcommand{\KFLT@af}{}%
\renewcommand{\KFLT@al}{}%
\renewcommand{\KFLT@as}{}%
\renewcommand{\KFLT@t}{}%
\renewcommand{\KFLT@textalign}{}%
\tdtextjustify%
\renewcommand{\KFLT@i}{}%
\renewcommand{\KFLT@lw}{}%
\setlength{\KFLT@w}{0pt}%
\setlength{\KFLT@h}{0pt}%
\renewcommand{\KFLT@s}{1}%
\renewcommand{\KFLT@r}{0}%
\setboolean{KFLT@f}{false}%
\setboolean{KFLT@ft}{false}%
\renewcommand{\KFLT@stretch}{1}%
\setlength{\KFLT@mo}{-1.2ex}%
\renewcommand{\KFLT@wp}{0}%
\renewcommand{\KFLT@va}{c}%
}
```

3.12 Row Start/End Processing

`\KFLT@maybestartfloatrow` Counts rows

After ending a preexisting row, move to the next row. The use of `\defcounter` makes this counter change local.

```
\newcommand*{\KFLT@maybestartfloatrow}{%
\KFLT@maybeendfloatrow%
\defcounter{KFLT@thiscol}{\value{KFLT@thiscol}+1}%
}
```

`\KFLT@maybeendfloatrow` Counts rows

Adds vertical space then resets to allow the start of a new row. The use of `\defcounter` makes this counter change local.

```
\newcommand*{\KFLT@maybeendfloatrow}{%
\ifthenelse{\cnttest{\value{KFLT@thiscol}}{>=}{\value{KFLT@numcols}}}%
{%

\addvspace{.75\floatsep}%

\defcounter{KFLT@thiscol}{0}%
}{}%
}%
```

3.13 Key Environment Helper Macros

`\KFLT@trackrows` Tracks and spaces rows and columns.

```
\newcommand{\KFLT@trackrows}
{%
```

If are nested inside a keyfloats or a subfloat:

```
\ifthenelse{%
\cnttest{\value{KFLT@keyfloatdepth}}>{0}%
\OR\boolean{KFLT@inkeysubfloats}%
}%
{% nested
```

Tracks row start and end:

```
\KFLT@maybestartfloatrow%
```

Possibly fill space between columns:

```
\ifthenelse{\cnttest{\value{KFLT@thiscol}}{>}{1}}%
{\hfill}{}%
}% nested
{}% not nested
}
```

`\KFLT@addtext` {⟨*{}* or *subgrp*⟩}

Adds optional additional text.

The argument is `{}` if a regular float, or `subgrp` if `\keysubfigs` or `\keysubtabs`.

```
\newcommand{\KFLT@addtext}[1]
{%
```

Is there text to add?

```
\ifcsempty{KFLT@#1t}%
{}% no text
{% text to add
{% local
```

Add some space, then create a full-width minipage to contain the text:

```
\unskip%
\addvspace{2ex}%
\begin{minipage}{\linewidth}%
```

Set the alignment and some text parameters:

```
\csname KFLT@#1textalign\endcsname%
\footnotesize%
\setlength{\parskip}{1.5ex}%
\setlength{\parindent}{0em}%
```

Typeset the actual text:

```
\csname KFLT@#1t\endcsname%
```

Close it all out with a little more space:

```
\end{minipage}%
\par\addvspace{2ex}%
}% local
}% text to add
}
```

`\KFLT@optionalname` {⟨*name*⟩}

Adds optional artist's name and the following space.

```
\newcommand{\KFLT@optionalname}[1]
{%
\ifthenelse{\equal{#1}{}}%
{}%
{#1~}%
}
```

`\KFLT@addartisttext` {⟨*{} or subgrp*⟩}

Adds optional artist's name and add'l text.

The argument is `{}` if a regular float, or `subgrp` if `\keysubfigs` or `\keysubtabs`.

One of two versions is used, depending on whether the tocdata package is available.

If tocdata is loaded, this float is a figure, and artist information is given, then the float's artist's information and optional text will be printed elsewhere by `\KFLT@caption`. Otherwise, it is printed here along with the text.

Two versions, depending on whether tocdata is loaded:

```
\@ifpackageloaded{tocdata}
{% tocdata loaded
```

If tocdata is loaded:

```
\newcommand{\KFLT@addartisttext}[1]
{%
```

Only use the artist name if this is a figure:

```
\ifthenelse{\equal{\csname KFLT@#1type\endcsname}{figure}}%
{% figure
```

Only use the artist name if a last name is given:

```
\ifcsempty{KFLT@#1al}%
```

A figure but no artist:

```
{\KFLT@addtext{#1}}%
```

A figure with an artist: will be handled by tocdata when the caption is created.

```
{}% fig w/ artist: text will be added by \captionartist in \KFLT@caption
}% figure
```

If not a figure, ignore artist information:

```
{\KFLT@addtext{#1}}%
}% KFLT@addartisttext
}% tocdata loaded
```

If tocdata is not loaded:

```
{% tocdata not loaded
\newcommand{\KFLT@addartisttext}[1]
```

```
458 {%
```

Only use the artist information if a last name is given:

```
459 \ifcsempty{KFLT@#1al}%
460 {}% last name not given
461 {% last name given
```

Add space and create the name inside a full-width minipage:

```
462 \addvspace{2ex}%
463 \begin{minipage}{\linewidth}%
```

If tocdata is not used, the artist's name is always centered:

```
464 \centering\footnotesize\textsc{%
465 \KFLT@optionalname{\csname KFLT@#1ap\endcsname}%
466 \KFLT@optionalname{\csname KFLT@#1af\endcsname}%
467 \csname KFLT@#1al\endcsname\csname KFLT@#1as\endcsname%
468 }%
469 \end{minipage}%
470 \par\addvspace{2ex}%
471 }% last name given
```

Any additional text follows the artist's name:

```
472 \KFLT@addtext{#1}%
473 }% KFLT@addartisttext
474 }% tocdata not loaded
```

Len `\KFLTimageboxwidth` The computed width of the object.

This may be used as the width parameter of a minipage to encase the object.

```
475 \newlength{\KFLTimageboxwidth}
```

Env `KFLT@boxinner` Typeset the contents in a width which depends on the keys.

```
476 \newsavebox{\KFLT@envbox}
477
478 \NewDocumentEnvironment{KFLT@boxinner}{}
479 {% keyboxinner
```

(Possibly) frame the contents of an `lrbox`:

```
480 \begin{lrbox}{\KFLT@envbox}%
```

Rotate the contents:

```
\turn{\KFLT@r}%
```

Box the contents in the width computed by `\KFLT@findwidths`:

```
\minipage{\KFLT@imagewidth}%
```

Spacing inside the box. Also default to regular justified text alignment.

```
\setlength{\parskip}{2ex}%
\renewcommand{\arraystretch}{\KFLT@stretch}%
}% keyboxinner
```

End of the environment:

```
{% endkeyboxinner
\endminipage%
```

End the rotated box:

```
\endturn%
```

Possibly frame:

```
\end{lrbox}%
\KFLT@frame{\usebox{\KFLT@envbox}}%
\par\addvspace{2ex}%
}% endkeyboxinner
```

`\KFLT@boxkeys` `{`⟨*keys*⟩`}` `{`⟨*figure/table*⟩`}` `{`⟨*lof/lot*⟩`}`

Default the options, adjust for a table, then parse the keys:

```
\NewDocumentCommand{\KFLT@boxkeys}{+m m m}
{%
\KFLT@defaults%
\renewcommand{\KFLT@type}{#2}%
\renewcommand{\KFLT@listtype}{#3}%
\setkeys{KFLT@keys}{#1}%
}
```

Env `KFLT@boxouter` `{`⟨*star?*⟩`}` `{`⟨*loc*⟩`}`

Boxes the contents of figures and floats.

Not used by subfigures.

```
\NewDocumentEnvironment{KFLT@boxouter}{m m}
{% boxouter
```

The `keyfigure` and `keytable` environments handle the contents in one of three possible ways, depending on whether it is called alone, inside a `keyfloats` environment, or inside a `keysubfigs` or `keysubtabs` environment.

Start the new subfigure or subtable, of the given width:

```
\ifthenelse{\boolean{KFLT@inkeysubfloats}}%
{\csname sub\KFLT@type\endcsname{\KFLT@rowboxwidth}}% subfloat
```

If `keyfloats`, place the contents inside a `minipage`:

```
{% not subfloat:
\ifthenelse{\cnttest{\value{KFLT@keyfloatdepth}}>{0}}%
{% keyfloats
\ifbool{KFLT@keywrap}
{\minipage[t]{\KFLT@rowboxwidth}}%
{\minipage[\KFLT@va]{\KFLT@rowboxwidth}}%
\captionsetup*{type=\KFLT@type}%
}% keyfloats
{% not keyfloats
```

Not a subfloat or `keyfloats`, so create a single float.

See if inside a `keywrap`. If so, force [H] and vertical align top.

```
\ifbool{KFLT@keywrap}%
{%
\par\addvspace{\baselineskip}%
\noindent\minipage[t]{\linewidth}%
\captionsetup{type=\KFLT@type}%
}%
{% not a keywrap
```

See if the float should [W]rap:

```
\ifthenelse{\equal{#2}{W}}%
```

Place [W], so create a wrapfloat from the wrapfig package:

```
{% [W]
```

Temporarily figure out `\KFLT@imagewidth`, and make the wrapped figure environment as wide as the desired image size plus frame:

```
\KFLT@findwidths%
```

```
\csname wrap\KFLT@type\endcsname{\KFLT@wp}%
{\KFLT@imagewidth+2\KFLTlooseframewidth}%
```

Change the interior image to the discovered fixed width.

```
\renewcommand{\KFLT@lw}{}%
\renewcommand{\KFLT@w}{\KFLT@imagewidth}%
}% [W]
{% not [W]
%
% See if the float should be positioned in the [M]argin:
%    \begin{macrocode}
\ifthenelse{\equal{#2}{M}}%
```

Place [M], so create a marginfloat:

```
{% [M]
\csname margin\KFLT@type\endcsname[\KFLT@mo]%
\captionsetup{type=\KFLT@type}%
}% [M]
{% not [M}
%
% See if the float should be positioned [H]ere:
%    \begin{macrocode}
\ifthenelse{\equal{#2}{H}}%
```

Place [H], so create an inline minipage:

```
{% [H]
\par\addvspace{\baselineskip}%
\noindent\minipage[\KFLT@va]{\linewidth}%
\captionsetup{type=\KFLT@type}%
}% [H]
```

Not [H], so create a float: For a starred float, make a two-column table in a two-col format.

```
{% not [H]
\IfBooleanTF{#1}%
{\csname \KFLT@type*\endcsname[#2]}{\csname \KFLT@type\endcsname[#2]}%
}% not [H]
}% not [M]
}% not [W]
}% not keywrap
}% not keyfloats
}% not subfloat
```

Handle a continued float. Ignored if in a subfloat.

```
\ifthenelse{\boolean{KFLT@cont}}{\ContinuedFloat}{}%
```

Figure out image and parbox widths for the contents:

```
\KFLT@findwidths%
```

If a table, place the caption above the contents:

```
\KFLT@captioniftype{table}{}%
```

Typeset the contents:

```
\center\unskip%
}% boxouter
```

End of the KFLT@boxouter environment:

```
{% endboxouter
\endcenter\unskip%
```

Optionally print artist's name and additional text:

```
\KFLT@addartisttext{}%
```

If a figure, typeset the caption below the contents:

```
\KFLT@captioniftype{figure}{}%
```

If are inside `keysubtabs`, end the subtable:

```
\ifthenelse{\boolean{KFLT@inkeysubfloats}}%
{
\csname endsub\KFLT@type\endcsname
}% subfloat
{% not subfloat
\ifthenelse{\cnttest{\value{KFLT@keyfloatdepth}}>{0}}% keyfloats?
{\endminipage}% keyfloats
{% not keyfloats
```

Not subfloat or `keyfloats`, so is an individual float.

Close the minipage or float:

See if in a `keywrap`:

```
\ifbool{KFLT@keywrap}{%
\endminipage%
\par\addvspace{\baselineskip}%
```

```
}
{% not keywrap
```

See if the float should [W]rap:

```
\ifthenelse{\equal{#2}{W}}%
```

Place [W], so close the wrap float:

```
{% [W]
\csname endwrap\KFLT@type\endcsname%
}% [W]
{% not[W]
%
% See if the float should be positioned in the [M]argin:
%    \begin{macrocode}
\ifthenelse{\equal{#2}{M}}%
```

Place [M], so close the marginfloat:

```
{% [M]
\csname endmargin\KFLT@type\endcsname%
}% [M]
{% not [M]
\ifthenelse{\equal{#2}{H}}%
{%
\endminipage% [H]
\par\addvspace{\baselineskip}%
}%
{% not [H]
\IfBooleanTF{#1}% starred float?
{\csname end\KFLT@type*\endcsname}{\csname end\KFLT@type\endcsname}%
}% not [H]
}% not [M]
}% not [W]
}% not keywrap
}% not keyfloats
}% not subfloat
}% endkeyboxouter
```

3.14 The `keyfigure` Environment

Env `keyfigure` * [⟨*loc*⟩] {⟨*keys/values*⟩}

```
\NewDocumentEnvironment{keyfigure}{s O{tbp} +m}
{%
\KFLT@boxkeys{#3}{figure}{lof}%
```

```
\KFLT@boxouter{#1}{#2}%
\KFLT@boxinner%
}%
{%
\endKFLT@boxinner%
\endKFLT@boxouter%
}
```

`Before keyfigure` Extra code to track rows outside of the `keyfigure` environment, before it starts. This is done to allow nesting without losing track of the prior level.

```
\BeforeBeginEnvironment{keyfigure}{%
\KFLT@trackrows%
}
```

3.15 The `\keyfig` Macro

`\keyfig` * [⟨*2: loc*⟩] {⟨*3: keys/values*⟩} {⟨*4: image filename*⟩}

A user-level macro to generate a figure with an image. This may be used by itself, or inside a `keyfloats` or `keysubfigs` environment.

```
\NewDocumentCommand{\keyfig}{s O{tbp} +m m}
{%
\KFLT@trackrows%
\KFLT@boxkeys{#3}{figure}{lof}%
```

After setting default values, override with the filename:

```
\renewcommand{\KFLT@i}{#4}%
\begingroup%
\KFLT@boxouter{#1}{#2}%
\KFLT@onefigureimage%
\endKFLT@boxouter%
\endgroup%
}
```

3.16 The `\keyfigbox` Macro

`\keyfigbox` * [⟨*loc*⟩] {⟨*keys/values*⟩} {⟨*box contents*⟩}

A user-level macro to generate a figure with arbitrary paragraph contents. This may be used by itself, or inside a `keyfloats` or `keysubtabs` environment.

```
\NewDocumentCommand{\keyfigbox}{s O{tbp} +m +m}
{%
\KFLT@trackrows%
\KFLT@boxkeys{#3}{figure}{lof}%
\begingroup%
\KFLT@boxouter{#1}{#2}%
\KFLT@boxinner%
#4%
\endKFLT@boxinner%
\endKFLT@boxouter%
\endgroup%
}
```

3.17 The `\keyparbox` Macro

`\keyparbox` * [⟨*loc*⟩] {⟨*keys/values*⟩} {⟨*box contents*⟩}

A user-level macro to generate a figure with arbitrary paragraph contents, but no number or caption. This is equal to a `\keyfigbox` with `cstar={}`. This may be used by itself, or inside a `keyfloats` or `keysubtabs` environment.

```
\NewDocumentCommand{\keyparbox}{s O{tbp} +m +m}
{%
\KFLT@trackrows%
\KFLT@boxkeys{#3}{figure}{lof}%
```

Force `cstar={}`:

```
\renewcommand{\KFLT@c}{}%
\setboolean{KFLT@cstar}{true}%
```

Continue like `\figbox`:

```
\begingroup%
\KFLT@boxouter{#1}{#2}%
\KFLT@boxinner%
#4%
\endKFLT@boxinner%
\endKFLT@boxouter%
\endgroup%
}
```

3.18 The `\keytab` Macro

`\keytab` * [⟨*loc*⟩] {⟨*keys/values*⟩} {⟨*tabular contents*⟩}

A user-level macro to generate a table with tabular contents. This may be used by itself, or inside a `keyfloats` or `keysubtabs` environment.

```
+
\NewDocumentCommand{\keytab}{s O{tbp} +m +m}
{%
\KFLT@trackrows%
\KFLT@boxkeys{#3}{table}{lot}%
\begingroup%
\KFLT@boxouter{#1}{#2}%
\KFLT@boxinner%
\centering%
#4%
\endKFLT@boxinner%
\endKFLT@boxouter%
\endgroup%
}
```

3.19 The `keytable` Environment

Env `keytable` * [⟨*loc*⟩] {⟨*keys/values*⟩}

```
\NewDocumentEnvironment{keytable}{s O{tbp} +m}
{%
\KFLT@boxkeys{#3}{table}{lot}%
\KFLT@boxouter{#1}{#2}%
\KFLT@boxinner%
\centering%
}%
{%
\endKFLT@boxinner%
\endKFLT@boxouter%
}
```

`Before keytable` Extra code to track rows outside of the `keytable` environment, before it starts. This is done to allow nesting without losing track of the prior level.

```
\BeforeBeginEnvironment{keytable}{%
\KFLT@trackrows%
}
```

3.20 A Row of Floats

`\KFLT@nonest` Error message if tried to nest subfloats.

```
\newcommand*{\KFLT@nonest}{%
\ifthenelse{%
\cnttest{\value{KFLT@keyfloatdepth}}>{0}%
\OR\boolean{KFLT@inkeysubfloats}%
}%
{%
\PackageError{keyfloat}{Cannot nest keysubfigs or keysubtabs.%
(Not in outer par mode.)}%
{The subcaption package do not support nested environments, so%
the keyfloat package cannot place a keysubfigs or keysubtabs%
environment inside another, or inside a keyfloats.}%
}%
{}%
}
```

Env `keyfloats` * [⟨*loc*⟩] {⟨*num columns*⟩}

User-level macro to create rows of figures/tables. Wrapping occurs after the number of specified columns. `keyfloats` environments may be nested to create a vertical set of figures next to a single larger figure, for example.

Place `\keyfig`, `\keyfigbox`, and `\keytab` commands inside the `keyfloats` environment.

Note that `lw` linewidth keys may need to be adjusted inside a `keyfloats`, `\keysubfigs`, or `\keysubtabs`, since `\linewidth` changes depending on the number of columns. Likewise, manually-selected `w` width and `h` tags may need to be adjusted to prevent overflow.

```
\NewDocumentEnvironment{keyfloats}{s O{tbp} m}
{%
```

Nest the environment:

```
\addtocounter{KFLT@keyfloatdepth}{1}%
```

If [H], nested, subfloats, or keywrap, use a minipage instead of a float:

```
\ifthenelse{%
\equal{#2}{H}%
\OR\cnttest{\value{KFLT@keyfloatdepth}}>{1}%
\OR\boolean{KFLT@inkeysubfloats}%
\OR\boolean{KFLT@keywrap}%
}%
```

Create an inline minipage:

```
{% [H] or nested
```

If nested, use different spacing as was computed in the outer nesting level:

```
\ifthenelse{%
\cnttest{\value{KFLT@keyfloatdepth}}>{1}%
\OR\boolean{KFLT@inkeysubfloats}%
}%
{\noindent%
\begin{minipage}{\KFLT@rowboxwidth}%
}%
{\bigbreak%
\noindent\begin{minipage}{\linewidth}}%
```

If inside subfloats, generate subfigures by default:

```
\ifthenelse{\boolean{KFLT@inkeysubfloats}}%
{}{\captionsetup*{type=figure}}%
}% [H] or nested
```

Isn't [H] or nested, so create a figure:

```
{% figure
\IfBooleanTF{#1}% starred figure, two-col figure in a two-col format
{\begin{figure*}[#2]}{\begin{figure}[#2]}%
}% figure
```

Compute the width of each entry:

```
\ifthenelse{%
\cnttest{\value{KFLT@keyfloatdepth}}>{1}%
\OR\boolean{KFLT@inkeysubfloats}%
}%
```

Nested or subfloats:

```
{\setlength{\KFLT@rowboxwidth}{.9\KFLT@rowboxwidth/\real{#3}}}%
```

Keyfloats:

```
{\setlength{\KFLT@rowboxwidth}{.9\linewidth/\real{#3}}}%
```

Center the contents:

```
\centering%
```

Count columns using `\defcounter` for a local effect:

```
\defcounter{KFLT@numcols}{#3}%
\defcounter{KFLT@thiscol}{0}%
}% starting keyfloats environment
```

When ending a `keyfloats` environment:

```
{% ending keyfloats environment
```

[H] or rows/subfigs? Close a minipage:

```
\ifthenelse{%
\equal{#2}{H}%
\OR\cnttest{\value{KFLT@keyfloatdepth}}>{1}%
\OR\boolean{KFLT@inkeysubfloats}%
\OR\boolean{KFLT@keywrap}%
}%
{\end{minipage}%
```

Spacing if nested:

```
\ifthenelse{%
\cnttest{\value{KFLT@keyfloatdepth}}>{0}%
\OR\boolean{KFLT@keywrap}%
}%
{}{\bigbreak}%
}% was [H]
```

Was not [H], so close a figure:

```
{% not [H]
\IfBooleanTF{#1}% starred figure?
{\end{figure*}}{\end{figure}}%
}% not [H]
```

Unnest the environment:

```
\addtocounter{KFLT@keyfloatdepth}{-1}%
}
```

`Before keyfloats` Extra code to track rows outside of the `keyfloats` environment, before it starts. This is done to allow nesting without losing track of the prior level.

```
\BeforeBeginEnvironment{keyfloats}{%
```

Track rows:

```
\ifthenelse{%
\cnttest{\value{KFLT@keyfloatdepth}}>{0}%
\OR\boolean{KFLT@inkeysubfloats}%
}%
{\KFLT@maybestartfloatrow}{}%
```

Possibly fill space between columns:

```
\ifthenelse{\cnttest{\value{KFLT@thiscol}}{>}{1}}%
{\hfill}{}%
}
```

3.21 Subfloats

`\KFLT@subgrpdefaults` Sets defaults before reading the keys.

```
\newcommand*{\KFLT@subgrpdefaults}{%
\setboolean{KFLT@subgrpcont}{false}%
\renewcommand{\KFLT@subgrpc}{}%
\setboolean{KFLT@subgrpcstar}{false}%
\renewcommand{\KFLT@subgrpsc}{}%
\setboolean{KFLT@subgrpscgiven}{false}%
\KFLT@setsubgrpfigure%
\renewcommand{\KFLT@subgrpl}{}%
\renewcommand{\KFLT@subgrpap}{}%
\renewcommand{\KFLT@subgrpaf}{}%
\renewcommand{\KFLT@subgrpal}{}%
\renewcommand{\KFLT@subgrpas}{}%
\renewcommand{\KFLT@subgrpt}{}%
\renewcommand{\KFLT@subgrptextalign}{}
\tdtextjustify
}
```

`\KFLT@subfloats` `{`⟨*starred?*⟩`}` `{`⟨*loc*⟩`}` `{`⟨*cols*⟩`}` `{`⟨*keys/values*⟩`}`

Start a subfloat environment

```
\NewDocumentCommand{\KFLT@subfloats}{m m m +m}
{%
```

Parse the key-value combinations:

```
\setkeys{KFLT@subgrpkeys}{#4}%
```

Nest the environment:

```
\setboolean{KFLT@inkeysubfloats}{true}%
```

Figure out the width of each subfloat. If starred, use the full-page `\textwidth`, else use `\linewidth`. .9 is used to leave a little room between columns.

```
\IfBooleanTF{#1}%
{\setlength{\KFLT@rowboxwidth}{.9\textwidth/\real{#3}}}%
{\setlength{\KFLT@rowboxwidth}{.9\linewidth/\real{#3}}}%
```

If [H], or in a `keywrap`, create an inline minipage:

```
\ifthenelse{%
\equal{#2}{H}%
\OR\boolean{KFLT@keywrap}%
}%
{%
\bigbreak\noindent\begin{minipage}{\linewidth}%
}%
```

Isn't [H], so create a float, possibly starred:

```
{%
\IfBooleanTF{#1}%
{\begin{\KFLT@subgrptype*}[#2]}{\begin{\KFLT@subgrptype}[#2]}%
}%
```

Set the caption type:

```
\captionsetup*{type=\KFLT@subgrptype}%
```

Process continued floats:

```
\ifthenelse{\boolean{KFLT@subgrpcont}}{\ContinuedFloat}{}%
```

Center the contents:

```
\center\unskip%
```

If this is a table, place the caption above the contents:

```
\KFLT@captioniftype{table}{subgrp}%
```

Not yet started a row of subfloats. The use of `\defcounter` makes these changes local.

```
\defcounter{KFLT@numcols}{#3}%
\defcounter{KFLT@thiscol}{0}%
```

Creat a group for the subfloats. Necessary in case they change `\tdtextcenter`, etc.

```
\begingroup%
}
```

`\KFLT@endsubfloats` {⟨*starred?*⟩} {⟨*loc*⟩}

Ends a subfloat environment.

```
\newcommand*{\KFLT@endsubfloats}[2]{%
```

End the group containing the subfloats:

```
\endgroup%
\unskip\endcenter%
```

A little extra space at the bottom:

```
\par\addvspace{\bigskipamount}%
```

Optionally print artist's name and additional text:

```
\KFLT@addartisttext{subgrp}%
```

If this was a figure, place the caption below the contents:

```
\KFLT@captioniftype{figure}{subgrp}%
```

End the float or minipage:

```
\ifthenelse{%
\equal{#2}{H}%
\OR\boolean{KFLT@keywrap}%
}%
{\end{minipage}\bigbreak}% was [H]
{% not [H]:
\IfBooleanTF{#1}% starred?
{\end{\KFLT@subgrptype*}}{\end{\KFLT@subgrptype}}%
}% not [H]
```

Unnest the environment:

```
\setboolean{KFLT@inkeysubfloats}{false}%
}
```

Env **keysubfigs** * [⟨*loc*⟩] {⟨*numcols*⟩} {⟨*keys/values*⟩}

A group of subfigures typeset in rows.

```
\NewDocumentEnvironment{keysubfigs}{s O{tbp} m +m}
{%
```

Error if trying to nest environments:

```
\KFLT@nonest%
```

Default the options:

```
\KFLT@subgrpdefaults%
```

Start of the environment:

```
\KFLT@subfloats{#1}{#2}{#3}{#4}%
}% the start of the environment
```

end of the environment:

```
{%
\KFLT@endsubfloats{#1}{#2}%
}
```

Env **keysubtabs** * [⟨*loc*⟩] {⟨*numcols*⟩} {⟨*keys/values*⟩}

A group of subtables typeset in rows.

```
\NewDocumentEnvironment{keysubtabs}{s O{tbp} m +m}
{%
```

Error if trying to nest environments:

```
\KFLT@nonest%
```

Default the options:

```
\KFLT@subgrpdefaults%
```

Default to table float type:

```
\KFLT@setsubgrptable%
```

Start of the environment:

```
\KFLT@subfloats{#1}{#2}{#3}{#4}%
}% the start of the environment
```

End of the environment:

```
{%
\KFLT@endsubfloats{#1}{#2}%
}
```

3.22 Margin Floats

Env `KFLT@marginfloat` [⟨*offset*⟩] {⟨*type*⟩}

```
\newsavebox{\KFLT@marginfloatbox}

\NewDocumentEnvironment{KFLT@marginfloat}{O{-1.2ex} m}
{% start
\FloatBarrier% keep floats in order
\begin{lrbox}{\KFLT@marginfloatbox}%
\begin{minipage}{\marginparwidth}%
\captionsetup{type=#2}%
\hbox{}\vspace*{#1}%
\noindent%
}% start
{\end{minipage}%
\end{lrbox}%
\marginpar{\usebox{\KFLT@marginfloatbox}}%
}
```

Env `marginfigure` [⟨*offset*⟩]

```
\ProvideDocumentEnvironment{marginfigure}{O{-1.2ex}}
  {\begin{KFLT@marginfloat}[#1]{figure}}
  {\end{KFLT@marginfloat}}
```

Env `margintable` [⟨*offset*⟩]

```
\ProvideDocumentEnvironment{margintable}{O{-1.2ex}}
  {\begin{KFLT@marginfloat}[#1]{table}}
  {\end{KFLT@marginfloat}}
```

Bool `KFL@keywrap` Tells the next keyfloat to wrap around some text.

```
\newboolean{KFLT@keywrap}
\boolfalse{KFLT@keywrap}
```

Len `\KFLT@keywrapwidth` The width of the object to be wrapped beside the text.

```
\newlength{\KFLT@keywrapwidth}
```

Len `\KFLT@keywrapparskip` The `\parskip` outside of the keywrap.

```
\newlength{\KFLT@keywrapparskip}
```

Len `\KFLT@keywrapparindent` The `\parindent` outside of the keywrap.

```
\newlength{\KFLT@keywrapparindent}
```

Env `keywrap` {⟨*width*⟩} {⟨*keyfloat*⟩}

```
\DeclareDocumentEnvironment{keywrap}{m +m}
{%
\par%
\setlength{\KFLT@keywrapwidth}{\linewidth}%
\addtolength{\KFLT@keywrapwidth}{-#1}%
\addtolength{\KFLT@keywrapwidth}{-2em}%
\minipage[t]{\KFLT@keywrapwidth}%
%
\setlength{\parskip}{\KFLT@keywrapparskip}%
\setlength{\parindent}{\KFLT@keywrapparindent}%
\booltrue{KFLT@keywrap}%
}
{%
\par
\endminipage%
\hfill%
\begin{minipage}[t]{#1}%
\booltrue{KFLT@keywrap}%
#2%
\par
\unskip\vspace{\smallskipamount}
\end{minipage}%
\par
}

\BeforeBeginEnvironment{keywrap}{
\setlength{\KFLT@keywrapparskip}{\parskip}
\setlength{\KFLT@keywrapparindent}{\parindent}
}
```

Change History and Index

Change History

Index

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

L

M

P

R

S

T

V

W